

ISSUE 18 | SEP 2008

Blender learning made easy

blender art

MAGAZINE

Baking AO Maps for Terrain Using Blender 3D

CNC Machining with Blender

MAKING OF: A Big City For A Game

Interview of Roland Hess

Landscapes,
Environments & Sets

COVERART - Jnaut - by Derek Watts

EDITORGaurav Nawani gaurav@blenderart.org**MANAGING EDITOR**Sandra Gilbert sandra@blenderart.org**WEBSITE**Nam Pham nam@blenderart.org**DESIGNER**

Gaurav, Sandra, Alex

PROOFERKevin Braun
Phillip Ryals
Bruce Westfall
Joshua Leung
Lynda Schemansky
Eric Pranausk
Noah Summers
Joshua Scotton
Mark Warren
Wade Bick
Patrick O'Donnell
Brian C. Treacy
Scott Hill
Henriel Veldtmann**WRITERS**Enrique Gonzales
Claudio "malefico" Andaur
SGMartínez & AARRuiz
Sandra Gilbert
Kernon Dillon
Michael Kersey
Dave Jarvis
Ken 'kat' Beyer**COVER ART**Jnaut - by Derek Watts
dwatts1@gmail.com

CONTENTS

Baking AO Maps for Terrain Using Blender 3D**9****Game Physics and Paths****14****CNC Machining with Blender****23****Create Scenes with a Seamless Background****32****MAKING OF: A Big City For A Game****39****MAKING OF: A Fantasy Landscape****44****MAKING OF: The Scene Of A Dream****59****Interviewed - Roland Hess****79**



Sandra Gilbert
Managing Editor

“The fall seems to inspire in quite a few of us is the need to organize and clean up”

September heralds summer's colorful slide into winter. The air gets crisp and cool after the stifling heat of summer and leaves everywhere start turning blazing shades of orange, gold and red. I have always reveled in the fact that the surrounding landscapes look like they have been painted in a riot of fiery colors. The beauty of Fall's colorful transformation has always been quite inspiring to me, so it's no wonder that the greater number of my projects find themselves started during the Fall season.

The inspiring nature of Fall not only kicks off my own yearly creative surge, it also lends itself very well to this issue's theme. This time around we are going to explore various methods for creating various landscapes and environments. Even if you haven't really done a lot of landscape work, between the beautiful Fall scenery and the inspiring articles in this issue, you just might find yourself overwhelmingly compelled to make a landscape or two yourself. In addition to landscapes, we also get an inside look into creating a custom Big City.

One other thing Fall seems to inspire in quite a few of us is the need to organize and clean up. Or at least it appears that way, as this is generally the time when most of us get stuck with a lot of yard clean up and organization. So when you need a break from the seemingly endless leaf raking, just set yourself right down and read all about “Project File Organization” and “How to set up Blender Libraries”.

So then you will be fully prepared for some overdue blender resource cleanup and organization. Just as soon as you finish raking that yard of course.

Happy Blending!

sandra@blenderart.org



*At some point or another
I have tried (or attempted)
just about all of them.*

Output Formats:

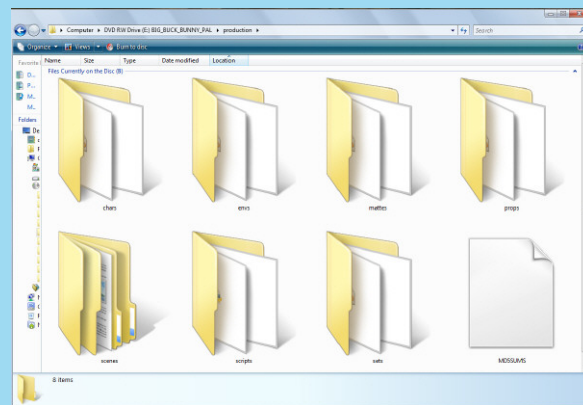
For years I have used the one folder organization method when working on a project. It worked out rather well, especially since each project had ten or so files (a couple of blend files, a few textures, maybe a reference image or two and the odd sound effect or song). Everything was in one place and very easy to find. So as things go, over the years my projects have gotten bigger, or more accurately, my project folders have gotten bigger. In fact, those folders seem to be collecting more and more files at a rather alarming rate. That makes it just a tad bit confusing when I want to go back and reuse something I have already done in a new project.

I hate to say it, mainly because I am so used to doing it my way, but I do believe it is about time to start using a more organized system (as well as set up a nice master Library, but that is a whole different project).

After a bit of poking around the blenderwiki, as well as peeking into all the folders and files on my copy of Big Buck Bunny, I am beginning to see some nice ideas for project organization. Now granted, I could go about organizing things any way I want, as long as I am consistent and can remember why I set it up that way to begin with. But I am looking for some pointers, and let's face it, these Blender gurus generally know what they are doing.

So let's take a look at how they set up their production files. When you open up the "production" folder, you see a series of folders.

They have set up individual folders for characters, environments, mattes, props, scenes, scripts and sets.



This seems to be a pretty good breakdown of all needed elements for an animation. At first I thought the scripts folder was for the story script, but it actually contains python scripts they used throughout the production. Further poking around revealed that the storyboards and animation scripts are in different folders on the disc. Most likely that was done just for the DVD set up, so I would probably go ahead and create a folder in the production folder for my animation script and storyboards, just to keep my project all together.

Okay, onward we go with our exploration. If you look into each of the folders you will notice that they are even further organized and separated into the various blends and folders needed for each folder type.

Let's take a closer look at what they included in each folder.

Characters:

- blend files: a separate file for each character in the animation
- textures (in its own folder): all textures for the characters
- python constraints (in its own folder): various python constraints that were written for the characters

Environments:

- blend files for various environmental elements (such as trees, rocks, flowers etc.)
- textures (in its own folder): all textures for the environmental elements

Mattes:

- blend files for the sky hill and clouds backgrounds
- textures (in its own folder)
- plates: exr files

Props:

- blend files for all the various props used by the characters (such as the apple, acorn, bow, etc)
- textures (in its own folder): all textures for the various props

Sets:

- blend files of the needed sets (think Hollywood, sets are much easier to build for only what the camera will see versus building the whole forest or mountain range)
- textures (in its own folder): all textures for the sets

Scripts:

- python scripts used during rendering
- sungrid: folder for sungrid sh files

Scenes: (we will talk more about these in a minute)

- each scene in the animation gets its own folder to contain the blends for that scene
- DVD folder that contains blends used for the credits
- elements folder: contains a blend for elements' animations

As you take a look at this list (or if you have the DVD or downloaded the production files), you will notice a few things. First, they created separate blend files for each element in the animation. This of course makes it much easier for various members of the team to work on various things at the same time by making use of Blender's ability to link to other blend files. That's a huge time saver.

Second, each folder has its own texture folder. Nothing is worse than having a large number of textures and not being able to remember right off hand what the texture goes to, although appropriate file naming would of course be helpful as well.

Third, depending on the folder, there are separate folders for python scripts and other needed elements.

In this particular setup, most of the folders and their contents actually end up being used as a “Master Library” for the production, with the individual scenes being the end point for all the various created elements. Each scene blend file is populated and linked back to all the needed elements for that scene. So the end result is that they have a very organized and efficient set of production files that can be modified as needed and updated automatically in the scene files. Which of course makes the production run smoother. But we already knew that, organizing always makes things more efficient and easier to use.

And as a bonus, once the production is over, it is now very easy to go back and actually reuse elements for a different project. Since every element has its own blend file, you can use the files as a Master Library to create a new animation or add the files to your own library and use them individually as needed. Which, quite honestly, beats my “one folder/one blend file” organization hands down.

All this poking around has shown me some good ways to improve my project organization and make it easier to not only complete my next project with less frustration, but to also be able to reuse elements later (well, once I get around to organizing all my previous projects into something resembling a useful Library, that is).

Blender News!



First Stop the Netherlands, Next the World

Ton has posted a rather amazing list of events (festivals and conferences) where Blender and Big Buck Bunny will be seen, and will be discussed over the next two months. Ton says that as of now he has stopped accepting invitations.

Here is a summary of his list of events (event, when, and a website for the event):

Netherlands Film Festival, official selection in main program

Utrecht, Sept 24 - Oct 3, 2008

<http://www.filmfestival.nl> Klik! Amsterdam animation festival

Amsterdam, 12-14 September 2008

<http://www.klikamsterdam.nl/>

IBC, International Broadcasting Conference

Amsterdam, 12-16 September 2008

<http://www.ibc.org>

Nederlandse Stripdagen (Dutch Comic days)

Houten, 27-28 September, 2008

<http://www.stripdagen.nl>

PICNIC'o8 (cross media conference)

Amsterdam, 24-26 September 2008

<http://www.picnicnetwork.org>

Cinekid, international film festival for the youth

Amsterdam, 18-26 October 2008

<http://www.cinekid.nl/>

Blender Conference

Amsterdam, 24-26 October 2008

<http://www.blender.org>

Stifo@Sandberg Conference

Amsterdam, October 31 2008

<http://www.all-media.info/page.php?id=65> (official announcement still pending)

Holland Animation Film Festival

Utrecht, 5-9 November 2008

<http://haff.awn.com/>

Specific details of how and when Blender and BBB will be shown at each event, are available at the [BBB website](#).

Yo Frankie! - Apricot Open Game Project



With the project all but wrapped up (they have announced an expected DVD release date for end of September), they have a released a new technical demo for us to play with.

Linux:

[Yo Frankie! Demo 1.1 - with GLSL support](#) - Linux 32bit [~27MB]

[Yo Frankie! Demo 1.1 - with GLSL support](#) - Linux 64bit [~27MB]

[Yo Frankie! Demo 1.1 - No GLSL](#) (Texture Face Mode) - Linux 32bit [~27MB]

[Yo Frankie! Demo 1.1 - No GLSL](#) (Texture Face Mode) - Linux 64bit [~27MB]

Windows:

[Yo Frankie! Demo 1 - with GLSL support](#)- [~31MB]

[Yo Frankie! Demo 1 - No GLSL](#) (Texture Face Mode) - [~31MB]

OSX:

[Yo Frankie! Demo 1 - No GLSL - OSX 10.4 Intel](#) [~32MB]

[Yo Frankie! Demo 1 - No GLSL - OSX 10.3 PPC](#) [~30MB]

Source Files: [Yo Frankie! Demo #1](#) - Source Files [25MB]

Blender Conference 2008



Dates: Fri-sat-sun, 24-25-26 October. Amsterdam, the Netherlands

The Seventh Annual Blender Conference will soon be kicking off in Amsterdam. In addition to the traditional activities enjoyed at each conference (discussions, courses, Suzanne Awards etc.), the Blender Institute will also be offering classes and courses the week before and after the conference for those interested.

Blender Institute training

The weeks before and the week after the conference, we will organize in the studio of the Blender Institute for two weeks of high quality training:

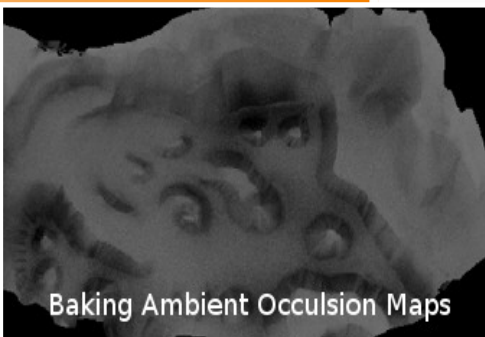
20th - 23th of October: Essential Blender

The Essential course is for people who already know 3D (other 3D programs, or some basics of Blender) and who want to get really into it.

27th - 31st of October: Advanced Blender

This course will focus on training artistic skills.

More information you can find [here](#).



Introduction

What are ambient occlusion maps?

Ambient Occlusion maps - or AO bakes, as they are more commonly known - are a useful way to generate 'shadows' for an object based on ambient light; this is the background and indirect lighting that illuminates objects in the world around us. For game and 3D media content, using ambient occlusion adds 'depth' and 'substance' to an object that may not otherwise be possible using other methods of in-game lighting. In 3D this can be represented by rendering a scene based ambient occlusion and 'baking' that to an image; the process tries to approximate the same non-directional lighting on objects in 3D space that happens in the real world by writing (baking) a gray scale image to memory (file) based on the physical characteristics of the model being processed.

How to bake ambient occlusion maps

It's quite simple to bake Ambient Occlusion (AO) maps using Blender, but there are a couple of things that need to be done in order for it to work correctly and yield the best possible results. The following tutorial has been written with Blender 2.46 in mind; the only major deviation from previous versions of Blender for AO baking is that UVW maps are now done in EDIT mode [TAB] instead of FACE EDIT mode [F] as previously.

Applying a material to the terrain mesh

The mesh needs a material and that material needs a filled texture slot. AO baking relies on the presence of a texture image when done in relation to game/3D content creation.

HOW TO : apply a material to the terrain mesh

Select the mesh first, then go in to the Shading buttons panel [F5] and either create a new material or edit the default one that's present.

Once done add a texture image slot by clicking on the "Texture buttons" icon [F6] and again either add a new slot or edit the default one present (should be named "Tex" if present). Browse to the image, select and add it, it will then appear in the texture slot preview window (shown bottom left of the image below).

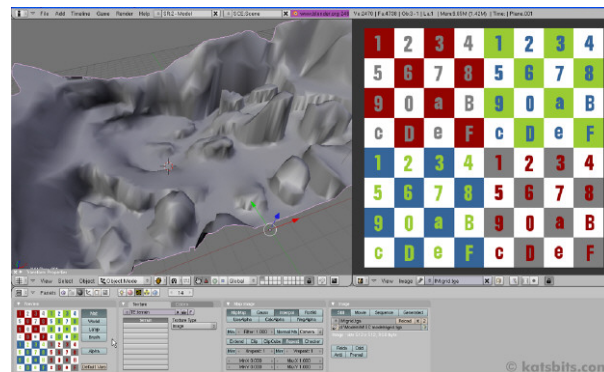


Fig. 1: The raw mesh ready to be set up for AO baking. A material is applied with a texture slot.

Displaying the texture applied to the mesh

By default all the triangles of the mesh will be in what's 'reset' (shown below). This is of no use for AO baking and needs to be fixed by creating or rebuilding the UV map so that it makes sense for the purpose of AO baking.

HOW TO : display the texture applied to the mesh (material)

To view the texture applied to the mesh press [Alt+Z], that will switch display mode to "Textured View" (Alt+Z toggles between "Flat Shading" and "Textured" views of an object in the 3D view-port).

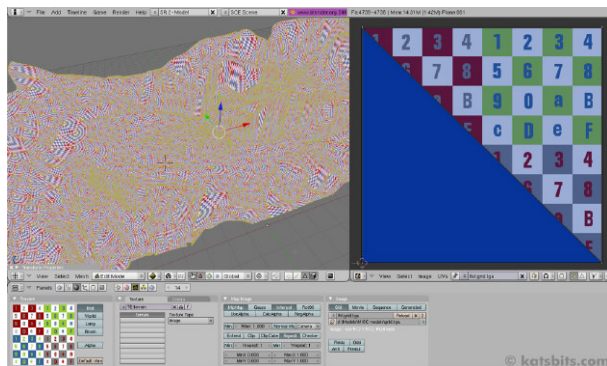


Fig. 2: In "Textured" view the UV map for the mesh is generated - UV mapping is now done in EDIT mode.

UV mapping the terrain mesh

Baking an AO map relies on a proper UV map being applied to the mesh. In Blender 2.46 this is all now done within EDIT mode.

HOW TO : UV map the terrain mesh

Press [TAB] to enter EDIT mode. Once there press [A] to select all (this may need to be done twice, once to clear isolated, selected faces and then again to reselect all faces). With all faces now selected press [U] to bring up the UV Calculation options pop-up. From here choose "Unwrap" to unwrap the mesh coordinates to a single UV map.

The result should be something similar to the image below. It's important to make sure here that the UV map stays within the bounds of the texture space; for best results make sure there are no stray UV vertices outside the edge of the texture.

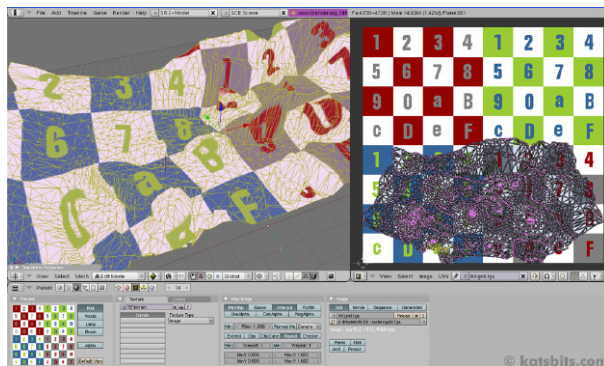


Fig. 3: The UV map is created, making sure that it stays within the boundary of the texture.

Baking an AO map using the default settings

The default setting for AO rendering will result in a gray scale image shaded to represent the physical characteristics of the mesh over which the texture is laid.

HOW TO : activating the AO settings

Click on the Shading [F5] icon to activate the additional relevant material buttons. Then click on the "World Buttons" icon to open up the panels associated with ambient occlusion.

Here you will find the "Amb Occ" tab and then click on the "Ambient Occlusion" button in that panel to turn on ambient occlusion; a series of buttons and sliders will then appear.

Leave everything 'as-is'.

Make sure the mesh is selected (keep the mouse in the 3D View) and then press [Ctrl+Alt+B] to start the ambient rendering process.



Fig. 4: Baking using the default settings can be done - [Ctrl+Alt+B] to initiate.

The rendering process will then replace and gradually update the place-holder image (checker image shown above) initially assigned to the material with the gray scale AO baked version. Depending on the complexity

and density of the mesh, this shouldn't take too long. The result will be a grainy grey scale image similar to below.

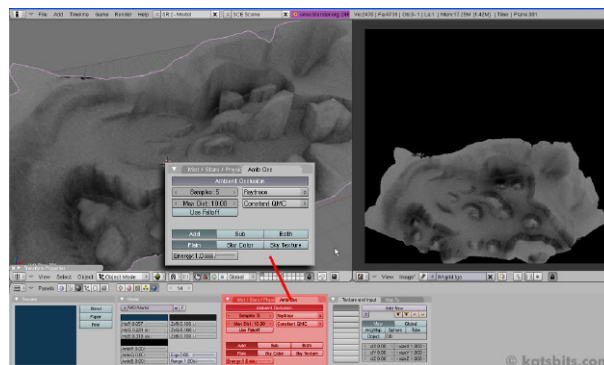


Fig. 5: This results in a grey scale image with highlights and shadows based on the depth and height of the mesh.

Colour tinting the AO baked image

The results of an AO bake can be colour influenced, to do so means switching to a different light influence system that effects the AO bake by using the colours assigned to the 'World' itself.

HOW TO : change the tint/colour of a baked AO map

Leave everything as it was for the previous render, except, click the button marked "Sky Colour." Nothing will change but the colour sliders in the "World" panel will now be active and influence the results of the AO baked image.

Change any of the "HoR", "HoG", "HoB" ("Horizon" RGB values) and/or the "ZeR", "ZeG", "ZeB" ("Zenith" RGB values) sliders to change the colour that will influence

the subsequent AO bake.

Press [Ctrl+Alt+B] to restart the AO render baking process again and watch the results.

In the test render shown below the colour influence was shifted to blue which resulted in a baked AO image that had a blue tint to higher elevations. Change colours appropriately as required. This has no effect on render time.

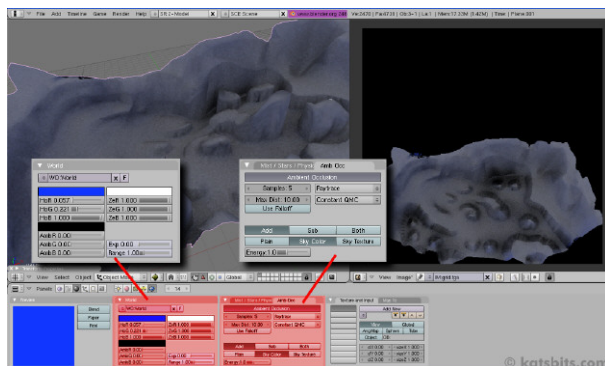


Fig. 6: Switching over to "Sky Color" which allows you to 'tint' the results. All other settings are left at defaults.

Quality settings for the AO baked image

Better quality results than those achieved using the default settings are possible by increasing the "Samples" value setting. Pre Blender 2.46 only 16 samples were available, however, 32 are now possible.

HOW TO : setting the samples value

Leave everything as is from the previous AO render and in the "Samples:" field either click on the ">" or "<" ar-

rows, or, LMB+Hold+Drag in the "Samples:" field to increase or decrease the number of samples used in the AO bake.

Press [Ctrl+Alt+B] to restart the AO bake process again.

The result of increasing the samples used is a much better quality image in terms of the amount of 'noise' it produces; the higher the setting the better the quality, but the slower the render time; using 16 or more sam-

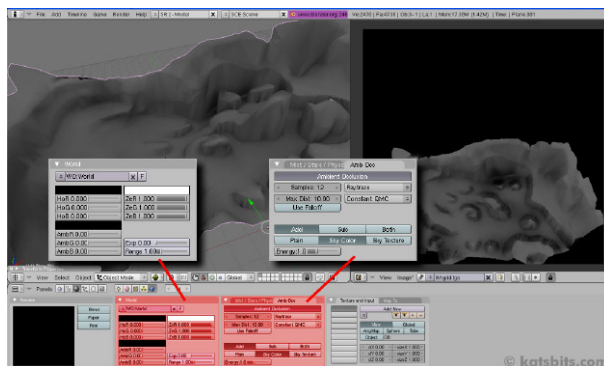


Fig. 7: Rendering a 'quality' AO bake means increasing the number of "Samples" - "12" gives acceptable results but samples do go up to 32. The more samples the longer the time required to bake the

ples results in significant amounts of time given over to the render process (relative to CPU speed).

Saving the baked ambient occlusion map

Once the ambient occlusion map has been baked to the level of fidelity that's required, it needs saving. It's best to use an image format that doesn't 'loss' compress the data in any way, so BMP, Raw TGA, and TIFF are all

viable alternatives; avoid using JPG wherever possible.

HOW TO : saving the final AO baked image

The resulting AO baked images needs to be saved from the "UV/Image Edit" view-port ([Shift+F10] if it's not visible). Click "Image" in the view-port 'header' bar and select "Save As..." (Image >> Save As.); the file browse view will open with a number of buttons and text fields.

In the header bar for this new window, look for and click on the drop-down option menu that displays an image 'type'; depending on the parent image (the image originally applied to the material and mesh) the drop down will be displaying "Targa", "Jpeg" or other 'title'.

Click and select one of the loss-less formats (tga, bmp, tiff, etc.) and then click "Save Image". The newly created baked ambient occlusion map will be saved to that location.

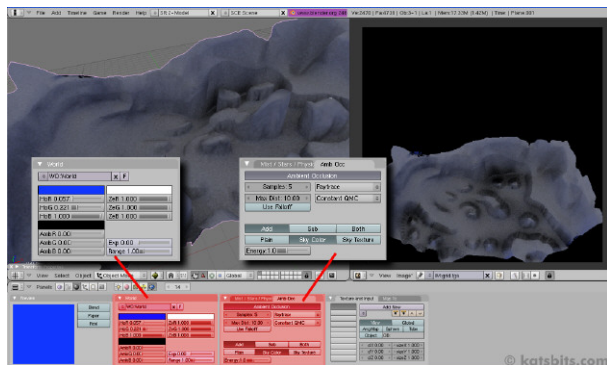


Fig. 8: Saving the ambient occlusion map to "RAW" tga. It's best to use an uncompressed or loss-less format for the resulting image when saving.

On saving the new AO map it will then become the active texture applied to the material and the UVW map of the mesh. ■

Info

Further 'advanced' ambient occlusion baking information (including video) can be found at the following URL;

http://www.katsbits.com/htm/tutorials/blender_ao_baking_terrain.htm

Additional 'terrain' tutorials can be found at;

Creating low poly terrain models

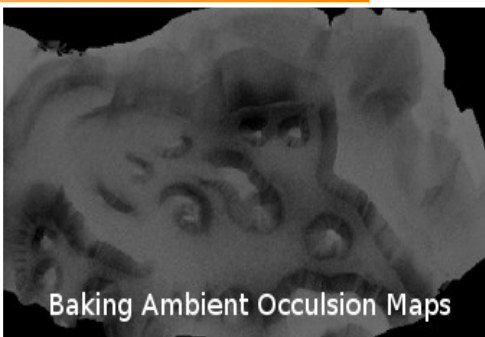
http://www.katsbits.com/htm/tutorials/blender_video_tutorial_terrain.htm

Rendering Skyboxes (including sample file)

http://www.katsbits.com/htm/tutorials/blender_rendering_skybox_environment.htm

Ken 'kat' Beyer

Website: KatsBits
URL: <http://www.katsbits.com>
e-mail: info@katsbits.com



Baking Ambient Occlusion Maps

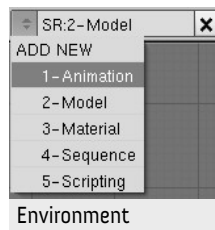
Introduction

This is a complex, advanced tutorial showing how to create an animation using Blender's Game Physics Engine and Paths. A Sphere will be dropped onto a Plane, moved along a Path, and then dropped onto a Box.

Blender has the ability to record physics-based actions to an object's IPO curve. This functionality allows a recorded physics sequence to be rendered as an animation. Since Paths control objects independently of physics, Path motion is not recorded without special intervention. Let us now intervene.

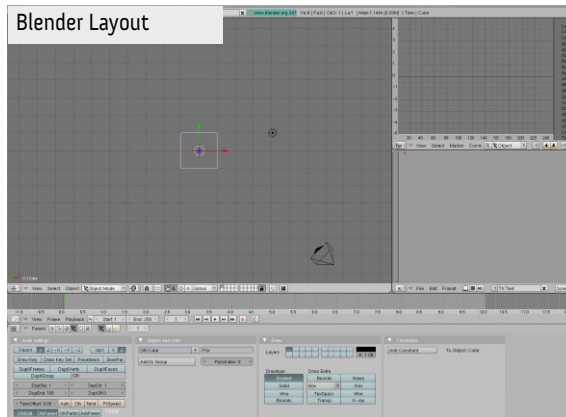
Setup

1. Press [Ctrl+X] for a new file.
2. Click OK to confirm.
3. Click SR:2-Model and change it to SR:1-Animation:



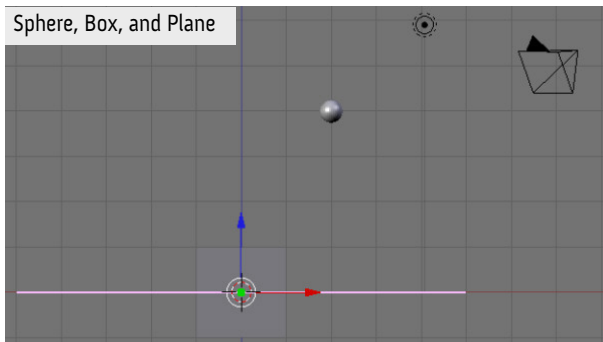
Layout Menu

4. Split the right-hand side of the 3D view window into two new windows.
5. Set the upper window to the IPO Curve Editor.
6. Set the lower window to the Text Editor.
7. Remove the Outliner window to increase the 3D view space, resulting in:



Objects

1. Add » Mesh » UVSphere
2. Click OK to accept the default options, then [Tab] to exit edit-mode.
3. Press [S], 0.25 then Enter to shrink the Sphere.
4. Press [G], [X], 2 then [Enter] to move the Sphere right.
5. Press [G], [Y], 2 then [Enter] to move the Sphere back.
6. Press [G], [Z], 4 then [Enter] to move the Sphere up.
7. Add » Mesh » Plane, then [Tab] to exit edit-mode.
8. Press [S], 5, then Enter to scale the Plane.
9. Press [NumPad1] for a front view:



10. Add » Curve » Path, then [Tab] to exit edit-mode.
11. Press [G], [Y], 2 then [Enter] to move the Path back.
12. Press [G], [Z], 0.25 then [Enter] to move the Path up.

This will align the Path's first vertex with the Sphere's center. After it falls, the Sphere will move along the Path. There are other ways to align vertices that don't rely on premeditated positioning, as in step 18.

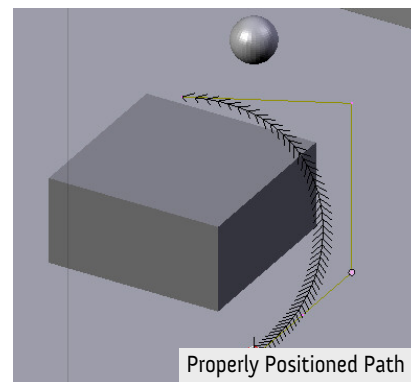
13. Object » Mirror » X Local
14. Press [NumPad7] for a top view.
15. Press [Tab] to edit the Path.
16. Press [A] to deselect all vertices.
17. Click RMB to select the last vertex (left-most).
18. Press [Shift+S] then click Selection -> Cursor.
19. Press [G], [Z], 1.5 then [Enter] to move the last vertex up.
20. Click RMB to select the second-last vertex.

21. Press [G], [X], 1 then [Enter] to align the second-last vertex with the last.

22. Press [G], [Z], 2 then [Enter] to move the second-last vertex up.

23. Click RMB to select the first vertex (right-most) and Press [Shift+S], then click Cursor -> Selection.

Rotating the scene should show a view similar to:



Materials

1. Press [Tab] to stop editing the Path.
2. Press [F5] for the Materials buttons.
3. Click RMB on the Cube to only select the Cube.
4. Under the "Links and Pipeline" panel, under "Link to Object", Rename Material to Cube.
5. Click RMB on the Plane to only select the Plane.
6. Under the "Links and Pipeline" panel, under "Link to Object", Click Add New.

7. Rename Material to Plane.

Record Path IPO

1. Press [NumPad7] for a top view.

2. Add » Empty

3. Click [SHIFT]+RMB on the Path.

The Empty and Path should both be selected.

4. Press [Ctrl+P] for the parenting menu.

5. Click Follow Path to parent the Empty to the Path.

6. Copy and paste the following code into the Text window:

```
#
# This script is used to map the IPO curve of an object as it travels
# through frames of an animation. The algorithm follows:
#
# 1. Calculate number of steps needed to animate at given framerate.
# 2. Get the selected object.
# 3. Duplicate the selected object (which will hold the generated IPO).
# 4. Reset the duplicate object.
# 5. Loop over the number of frames at the calculated framerate.
# 6. Set the active frame.
# 7. Position the duplicate object to the same place as the selected one.
# 8. Copy the selected object's location and rotation into the duplicate's
#    IPO.
#
# Usage:
# (1) Select the object that follows a curve.
# (2) Load this script in Blender's Text window.
# (3) Move the mouse to the Text window.
# (4) Press Alt-p.
#
import Blender
```

```
framesPerSecond = Scene.GetCurrent().getRenderingContext().fps
firstFrame      = 0
lastFrame       = 100
stepsPerFrame   = (lastFrame - firstFrame) / framesPerSecond
selected = Object.GetSelected()[0]
Object.Duplicate()
duplicate = Object.GetSelected()[0]
duplicate.clearParent()
duplicate.clearIpo()
for frame in range( firstFrame, lastFrame, stepsPerFrame ):
    Blender.Set( 'curframe', frame )
    duplicate.setMatrix( selected.getMatrix() )
    duplicate.insertIpoKey( Object.LOC )
```

7. Change the name of the script to copy-path.py.

8. Save the script as copy-path.py.

9. Click RMB to select the Empty.

10. Move the mouse to the Text window.

11. Press [Alt+P] to execute the script.

12. While executing, the script instructs Blender to scrub through frames at 25 frames per second (default). With 100 frames, only every fourth frame is used. Each frame that is used causes the duplicate object to have its location and rotation recorded as an IPO key (i.e., a vertex in that object's IPO curve).

13. Move the mouse to the 3D window.

14. Press [Del] then [Enter] to delete Empty.001.

15. Click RMB to select the Empty.

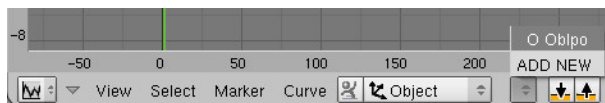
16. Press [Alt+P].

17. Click Clear Parent to remove the Path as the Empty's parent.

18. Press [Shift+DownArrow] to Reset the IPO to frame 0.

If the Empty moves in unexpected ways, it is likely this step was missed.

19. In the IPO window, set the Empty to use the newly



created IPO data block, Obipo.

Empty's New IPO

20. Click RMB to select the Path.

21. Press [M], 2, then [Enter] to move the Path to another layer.

22. Press [F7] for the Object buttons.

23. Click RMB to select the Empty.

24. Under the "Animation Settings" panel, Set TimeOffset to 180.

This is the number of frames the Sphere takes to hit the Plane.

Game Engine Actors

1. Press [F4] for the Logic buttons.

2. Click RMB to select the Sphere.

3. Click Actor.

4. Click Dynamic.

5. Click Bounds.

6. Set Radius to 0.75.

7. Change Box to Sphere (as the bounds type).

8. Click RMB to select the Plane.

9. Click Actor.

10. Click Bounds

Game Engine Logic

When the Sphere collides with the Plane the following must happen:

- The Sphere makes the Empty its parent.
- The Empty plays its IPO.
- The Sphere's IPO keys are recorded for each frame.

Logic Blocks - Empty

1. Click RMB to select the Empty.

2. Click Add Property.

3. Change the name to collision.

4. Change the type from its default value Float to Bool.

This will record the Sphere colliding with the Plane. Upon collision, the Sphere must follow the Empty (which has an IPO that was derived from the Path that was banished to Layer 2).

5. Under the "Sensors" Panel, Click Add (right next to Empty) to add a sensor to the Empty.

6. Change the name of the sensor to s.copy.

This sensor will help continually copy the value of the Sphere's collision property.

7. To add another sensor, Click Add, below Sensors.

8. Change the sensor type from Always to Property.

9. Change the name of the sensor to s.ipo.

10. Set the value of Prop to collision.

11. Set the value of Value to True.

12. Under the "Controllers" panel, Click Add, to add a controller.

13. Change the name of the controller to c.copy.

14. To add another controller, Click Add, below Controllers.

15. Change the name of the controller to c.ipo.

16. Under the "Actuators" panel, Click Add, to add an actuator.

17. Change the name of the actuator to a.copy.

18. Change the actuator type from Motion to Property.

19. Change the actuator's property type from Assign to Copy.

20. Set the value of Prop to collision.

21. Set the value of OB to Sphere.

22. Set the value of Prop to collision.

23. To add another actuator, Click Add, below Actuators.

24. Change the actuator type from Motion to Ipo.

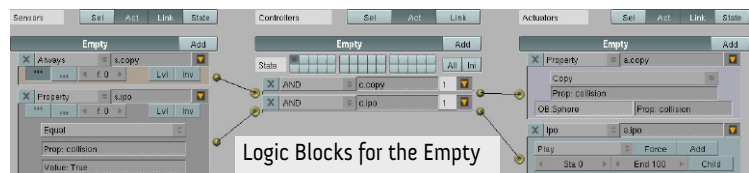
25. Change the name of the actuator to a.ipo.

26. Set the value of End to 100.

27. Link Sensor copy to Controller copy to Actuator copy.

28. Link Sensor ipo to Controller ipo to Actuator ipo.

The logic should look similar to:



The logic block instructs Blender to continually copy the Sphere's collision property into the Empty's property of the same name. Once the collision property is set to True, the Empty is instructed to replay its IPO. All that remains is to set the Sphere's parent to the Empty, remove the parent relationship when appropriate, and record the Sphere's complete IPO curve.

Logic Blocks - Sphere - Sensors

1. Click RMB to select the Sphere in the 3D Window, then Click Add Property.

2. Change the name to collision.

3. Change the type from its default value Float to Bool.

4. Click Add Property.

5. Set the name to time.

6. Change the type from Float to Timer.

A timer will be used to remove the Sphere's parent (the Empty). It is also used to calculate the current frame.

7. Under the "Sensors" Panel, Click Add, below Sensors.
8. Change the name to s.plane.
9. Change the sensor type to Collision.
10. Toggle M/P on.
11. Set the value of M/P to Plane.
12. Click Add, below Sensors.
13. Change the name to s.time.
14. Change the sensor type to Property.
15. Toggle Activate TRUE level triggering on.
16. Change the type from Equal to Interval.
17. Set the value of Prop: to time.
18. Set the value of Min: to 4.0.
19. Set the value of Max: to 4.1.
20. Click Add, below Sensors.
21. Change the name to s.start.
22. Toggle Activate TRUE level triggering off.
23. Click Add, below Sensors.
24. Change the name to s.record.
25. Set the value of f: to 2.
26. Toggle Activate TRUE level triggering on.

27. Click Add, below Sensors.

28. Change the name to s.stop.

29. Change the sensor type to Collision.

30. Toggle M/P on.

31. Set the value of M/P to Cube.

Logic Blocks - Sphere - Controllers

1. Under the "Controllers" Panel, Click Add, below Controllers, five times.

2. Change the name of the controllers, from top to bottom as:

- c.plane
- c.time
- c.start
- c.record
- c.stop

3. Change the default controller type from AND to Python for c.start and c.record.

4. Create a new text file in the Text window by clicking the arrows next to the datablock name and clicking ADD NEW.

5. Copy and paste the following code into the Text window:

```
#
# This script is used to record the IPO curve of an object as it is
# assaulted by the Game Physics engine. It should be executed automati-
# cally
# via Logic Blocks. The algorithm follows:
#
# 1. Get the object being controlled by physics.
# 2. Create a new IPO for recording the motion of that object.
# 3. Store the IPO curves for use by the frame recorder.
#
import Blender
# Get the name of the object being controlled by physics.
#
gameObject = GameLogic.getCurrentController().getOwner()
objectName = gameObject.getName()[2:]
# Create an IPO of type Object named 'Recorded IPO'.
#
blenderObject = Blender.Object.Get( objectName )
ipo = Blender.Ipo.New( 'Object', 'Recorded IPO' )
blenderObject.setIpo( ipo )
# Get the position of gameObject as IPO curves.
#
locx = ipo.addCurve( 'LocX' )
locy = ipo.addCurve( 'LocY' )
locz = ipo.addCurve( 'LocZ' )
# Keep a reference to the curves in a global variable. The variable
# (GameLogic.rec) is used by a script that records the position of the
# game Object per frame.
#
```

6.Change the name of the script to record-ipo.py.

7.Save the script as record-ipo.py.

8.Create a new text file in the Text window by clicking the arrows next to the datablock name and clicking ADD NEW, and then change the name to record-frame.py and save it as record-frame.py.

9.Copy and paste the following code into the Text window:

```
#
# This script is used to record the position of an object as it travels
# through each frame, independent of the Game Physics. If Blender is set
# to record Game Physics to IPO, toggle that menu item OFF. This is a
# replacement. The algorithm follows:
#
# 1. Get the object being controlled by physics.
# 2. Get the object's position (for a given frame).
# 3. Update the recorded curves (stored in the global variable
# GameLogic.rec).
#
import Blender
from Blender import Scene
gameObject = GameLogic.getCurrentController().getOwner()
position = gameObject.getPosition()
frame = gameObject.time * Scene.GetCurrent().getRenderingContext().fps
# Only try to update the curves if the variable has been initialized.
#
if hasattr( GameLogic, 'rec' ):
# Set the curves for the X, Y, and Z axis.
#
GameLogic.rec[1].addBezier( (frame, position[0]) )
GameLogic.rec[1].update()
GameLogic.rec[2].addBezier( (frame, position[1]) )
GameLogic.rec[2].update()
GameLogic.rec[3].addBezier( (frame, position[2]) )
```

10.Under the "Controllers" Panel, Set the value of Script: for controller c.start to record-ipo.py.

11.Likewise, Set the value of Script: for c.record to record-frame.py.

12.Under the "Actuators" Panel, Click Add, below Actuators, five times.

10. Under the "Controllers" Panel, Set the value of Script: for controller c.start to record-ipo.py.

11. Likewise, Set the value of Script: for c.record to record-frame.py.

12. Under the "Actuators" Panel, Click Add, below Actuators, five times.

13. Change the type of actuators, from top to bottom as:

- Property
- Parent
- Property
- Parent
- Game

14. Change the name of the actuators, from top to bottom as:

- a.start.parent
- a.parent.empty

• a.stop.parent

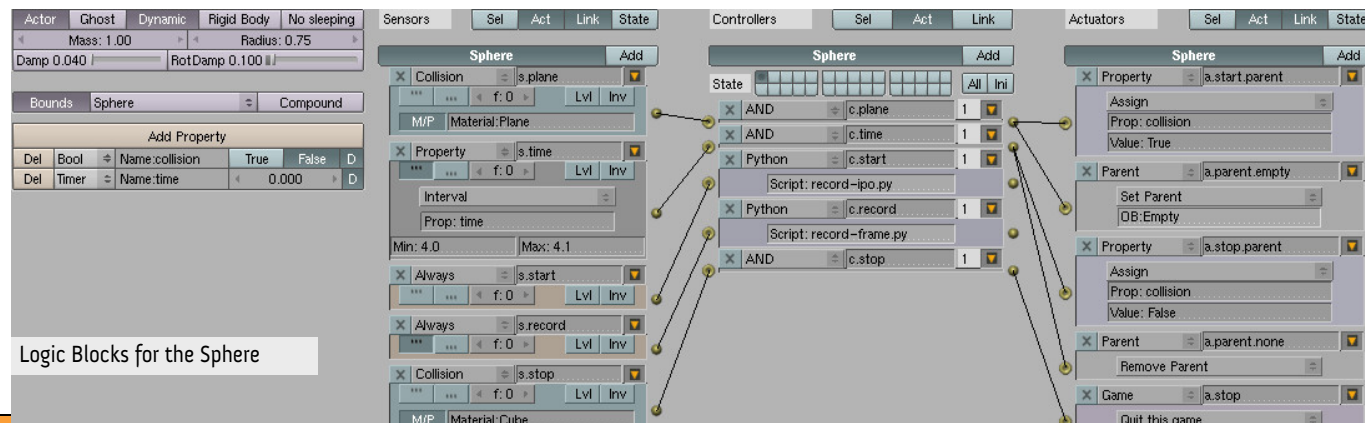
• a.parent.none

• a.stop

15. From top to bottom:

- Set the Prop value of "a.start.parent" to collision.
- Set Value of "a.start.parent" to True.
- Set the OB value of "a.parent.empty" to Empty.
- Set the Prop value of "a.stop.parent" to collision.
- Set Value of "a.stop.parent" to False.
- Change Set Parent of "a.parent.none" to Remove Parent.
- Change Start new game of "a.stop" to Quit this game.

16. Link Sensors to Controllers to Actuators:



Render Animation

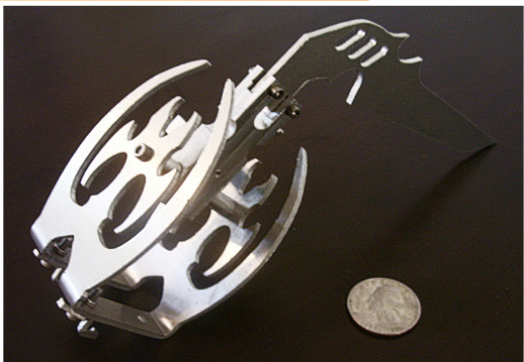
1. Move the mouse cursor over to the 3D window.
2. Press [P] to invoke the wrath of the Game Physics.
3. Press [Alt+A] to play the animation using IPO curves.

The Sphere should fall from gravity to the Plane, follow the Path, then fall from gravity to the Box.

4. Render » Render Animation. ■

David Jarvis

URL: <http://www.davidjarvis.ca/>



Introduction

In this guide I will walk through the stages of modeling in blender to the bringing it to life using a CNC Mill. This guide ONLY covers 2.5D cutouts. (AKA Profiling) Full 3D CNC will be covered in a later guide if there is enough interest.

Before you start

Now some of you will be thinking as you read this, "Why the heck don't you use and open source CAD program instead of all this!" The answer is simple really, people wanna know how, and I am providing one possible methodology. 'nuff said? This guide is going to assume you are very well versed in Blender itself. If you are a novice, I assure you, this will be Gr33k to you. Well before you get to the point where you are making some kewl guy thing-a-ma-jigs with your CNC machine, you'll need to start at the basics. Blender does not lend itself to CNC very well as that is not its intended function. It's just something else you can do with it, if you want to invest the money and time. Money? Ya, Blender may be free, but CNC isn't. So before you delve into this guide, you can expect to fork out no less than \$500.00 to get the bare bones minimum.

A CNC Mill or Router can easily be built and there are dozens of sites with plans for CNC Routers and schematics for CNC controllers, etc, etc. in addition to your hardware, you'll need software. Yes, Blender isn't going to run your mill for you. I don't want to dwell too much on the costs, but thought it prudent to give fair warning before you read through this

guide and realized that you can't afford to get into CNC. So, I will just list the components you will need: **Blender: FREE!**

Your modeling software! (With a few free plug-ins to fill out the edges.)

DXF Exporter script: (A HUGE thanks to Migius, Yorik, and Stani for creating this!) **\$0**

Download the script from here:
http://wiki.blender.org/index.php/Scripts/Manual/Export/autodesk_dxf

This script has a number of features to export out a flat DXF file For the purposes of this document, we will only be covering the default settings.

NOTE: As of this writing, this script is still in Beta, but is 100% functional for our purposes.

CAM Software: (Computer Aided Manufacturing) \$50 to \$1,000+

CAM software is the middle ware between the CAD world and the manufacturing world. In a nutshell, CAM software translates the object you have created (dxf file) into machine code called G-Code. G-Code defines the Mill/Router/Lathe's toolpath which the machine will use to create your object.

Recommended software:

SheetCam - <http://www.sheetcam.com/>

Cut3D - <http://www.vectric.com/>

MeshCam - <http://www.grzsoftware.com/>

By Michael Kersey

CNC controller: \$50 (building your own from a schematic) to \$10,000+

This is the device that translates the G-Code being fed to it by the CNC Controller software into pulses that the Stepper motors or Servo motors use to physically move the machine's cutting device.

Recommended:

Xylotex - <http://www.xylotex.com/>

Geckodrive - <http://www.geckodrive.com/>

Various schematics for making your own - <http://www.discovercircuits.com/S/stepper.htm>

CNC Machine: \$250 to \$500,000+

Here, there are so many choices it is mind boggling. Mills (Generally used on Metal), Routers (Generally used on woods and softer materials) Lathes, Laser cutters, Plasma cutters, Sewing machines, Engravers, Plotters, Abrasive Waterjets and even 3D printers! Most people who are building their own CNC machines make CNC Routers. There are dozens for sale that are relatively inexpensive and hundreds more plans to make you own from scratch.

Recommended: Build it yourself!

<http://buildyourcnc.com/>

<http://www.cnccookbook.com/>

<http://www.cnc411.com/>

<http://www.cncfuture.com/>

<http://www.cncinformation.com/>

<http://fabathome.mae.cornell.edu/>

So, now that you have an idea of what you are getting yourself into, and you're still determined to proceed, then good for you!

Modeling

There are two distinct approaches to modeling when it comes to CNC. 2.5D and 3D. 3D is easy enough, a 3 dimensional object that is cut from a solid block of material. 2.5D is more inline with Engraving machines, Plasma cutters, Laser cutters, and Waterjets. In 2.5D the objective is more to 'cut out' than to 'Carve out' the object. Since it turned out to be far more difficult for me, and it is what I was after with my CNC machining, I'll cover 2.5D first. Trust me, after 2.5D, 3D is a cakewalk.

2.5D Modeling

For the intent and purposes of what I was trying to create with my CNC Mill, I will explain 2.5D modeling in blender from that point of view. This is by no means a definitive guide to the limit of all the possibilities, so feel free to let your noodle wander as you are reading.

So, let's say...

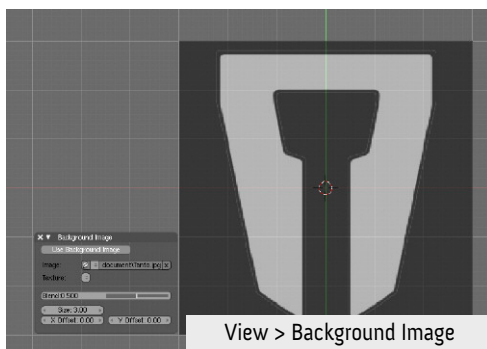


You wanted to cut something out of a piece of metal shaped roughly like the above image. For the point of illustration, we will use a sheet of 1/16" aluminum as our material. Now, I'm sure all my fellow Blender fanatics out there would scoff and say, "Hell, I could model that in under a minute!" Not so fast hot shots, remember the goal... I will repeat this until you are absolutely sick of hearing it. Blender is NOT a CAD program. Hence, when you are modeling something in blender with the intent of using it in a CNC application, you need to let go of several preconceived notions about 'how to model'. When modeling for the purpose of a 2.5D CNC object, you need to concentrate more on how to tell the CNC machine what you want it to do, than making the part in 3D, and then figuring out how to go from there. (I made that mistake)

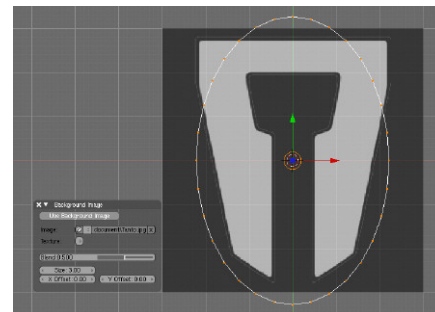
The object is best thought of in this case, just like it is, printed on a piece of paper. How would you cut it out? Well, you would use a pair of scissors or a Xacto knife and cut around the edges, right. Good. Now you are starting to follow me.

In Blender, what you are trying to create is not so much the part itself, but the toolpath required to cut it out. Relax, it's easier than most things people do in Blender!

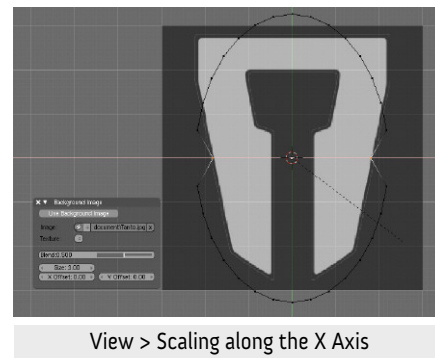
First you need to take your picture, and put it in as a background image in the Blender 3Dview. Like so:



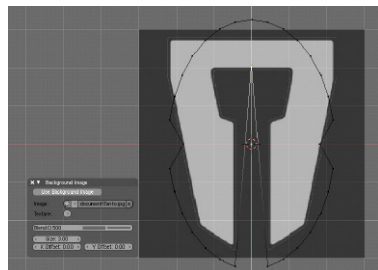
Now, we are going to create an 'outline' of the object that will be translated into the toolpath for your CNC machine. To start, create a simple circle [SPACE ? Add ? Mesh ? Circle] in the top view [Num 7]. And scale [S+X and S+Y] it on the X and Y axis's so it roughly matches the outer edge of your object. (See next image)



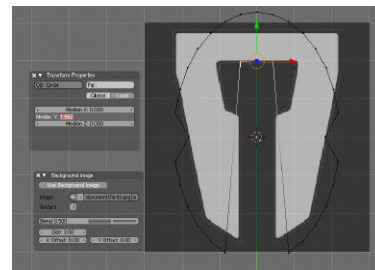
Now, for parts that are that are going to be symmetrical in nature, it is imperative to make sure as you are modeling that you keep you Vertices's aligned. Instead of grabbing each vertex and putting it in place, grab them in pairs. One on each side, and scale them along the Axis's into place. I.e. [S] to scale, and then [X] to constrict the scaling function to the X axis. The X and Y axis should be the only ones being utilized in the top view. (There is a method behind the madness here, using the top view will save you time during the later steps when converting this to the final product.)



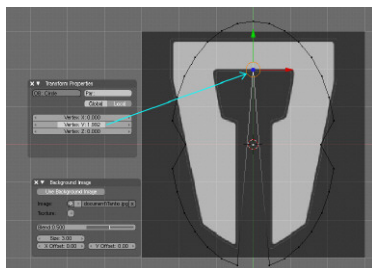
For the non symmetrical areas, it isn't as important to be 'dead on' but, if you want the final product to be as pristine as it can be, always move the vertices using the Axis constraints.



Now that you have them, move them quickly into place by changing the Median Y to the Y Axis of the first Vertex. In the case of our example, 1.992.

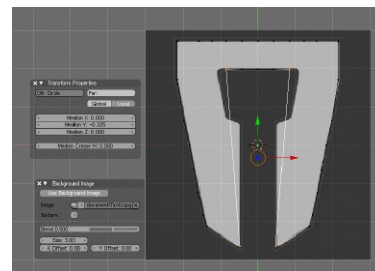


You'll find in most objects you create that there are straight pieces that are difficult to get perfectly straight, in these cases, use of the Transform Properties [N] will really speed along your modeling. First, note the Vertex location on what ever axis you are trying to make a straight line on. (See image on right).

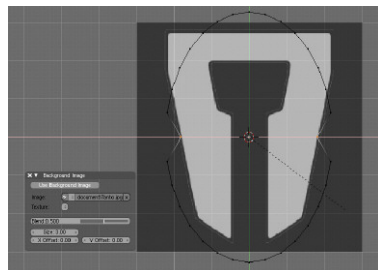


poink Your Vertices pop perfectly into place! So you have the basics, wash, rinse, and repeat...

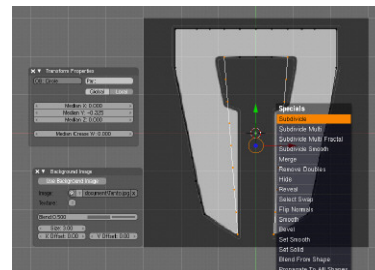
When you find that you have more 'Corners' than vertices to wrap around them (See image on right).



In this Example, the Vertex is at 1.992 on the Y axis. So, grab another opposing pair... Note: Only grab two vertices at a time when using this method. Remember, when more than one Vertex is selected in edit mode, the Transform apply to the Median of ALL the selected Vertices.



Just highlight the opposing vertices (Remember to keep symmetrical!) and subdivided them a few times. (See image on right)

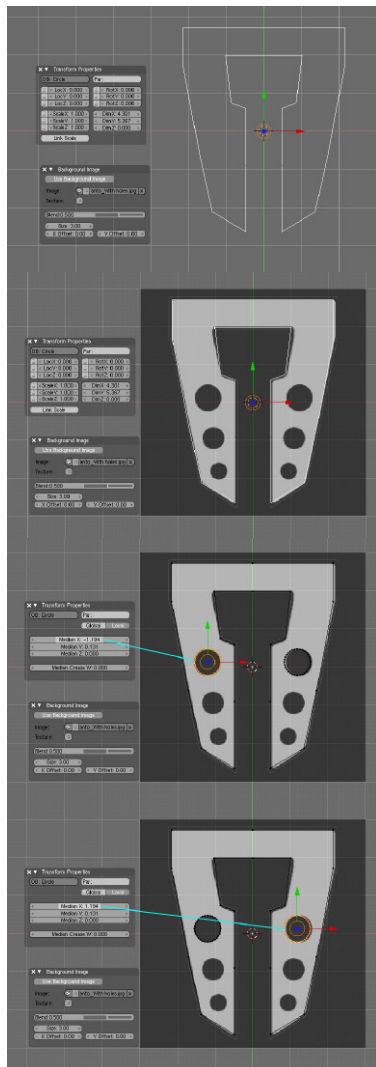


Ready to keep modeling!

Looking good... But that's a little plain. Certainly we can do a little better. Let's add some of those kewl looking structural holes to make this a little more interesting.

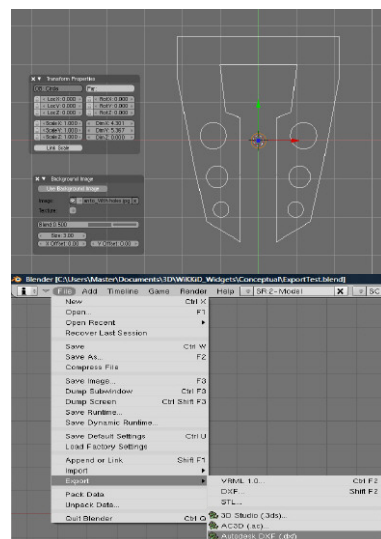
Ya, that looks allot more better To add the holes on the model, In edit mode, just add circles, scale them to size, and move them into place. Remember to keep your symmetry correct.

The image file may not be 100% symmetrical. So use the Transform Properties to keep your model correct. If the Median is -1.194 on the left...



Make sure that the Median is 1.194 on the right. And when you're finished...

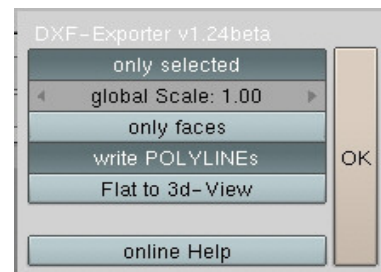
Your model will be perfectly symmetrical! So we are ready to export the mesh to a DXF file using DXF exporter script listed in the beginning of this document.



Export your model to DXF using the Autodesk DXF (.dxf) script as pictured above.

NOTE: Do NOT use the built in DXF Export. The built in DXF Export creates a valid DXF file using Polyfaces instead of Polylines. Several CAM programs do not support Polyfaces, including the current version of SheetCAM (Which we are using here to demonstrate the process.)

Use the default settings from the script.



From CAD to CAM... and Beyond!

Note: In SheetCam, you will have to define the tool, material, machine, etc, etc. which is not covered in this guide. (Perhaps later if enough people want to know but I would suggest hitting SheetCAM's forum first)

Once you have SheetCam properly set up, you can import your Model and see it represented on the material it is to be cut from for the first time!

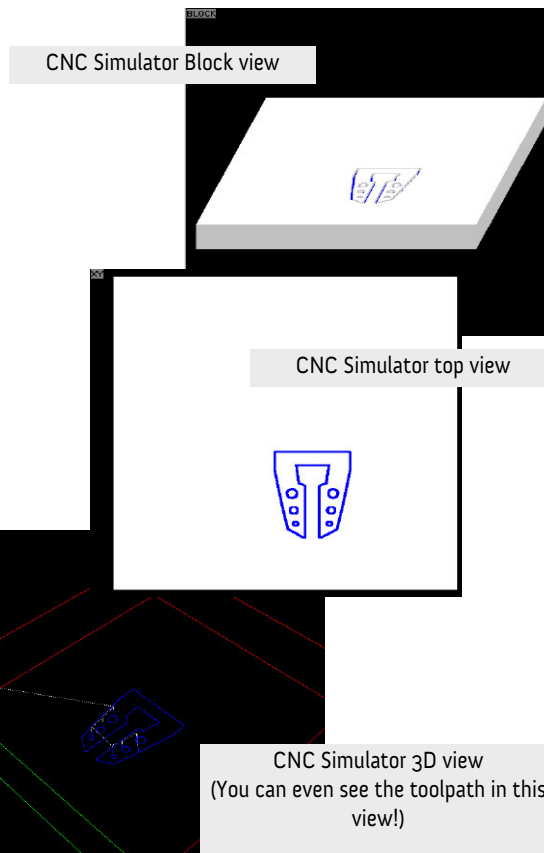
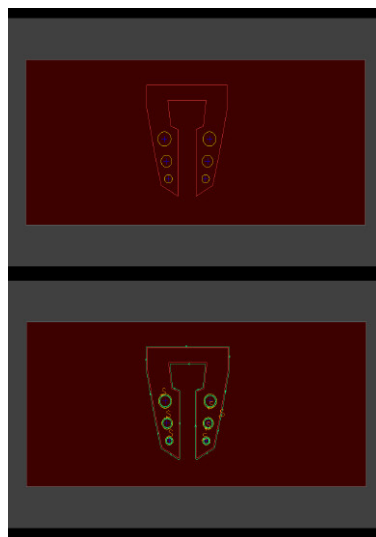
In SheetCam, the Grey area represents the machines cutting area. The Red (Burgundy for you Hue aficionados) represents the Material. Black is the background, and your hella kewl guy part is easy to spot!

You set up the Contour job to cut out your Object, and SheetCAM generates the toolpath based on your model!

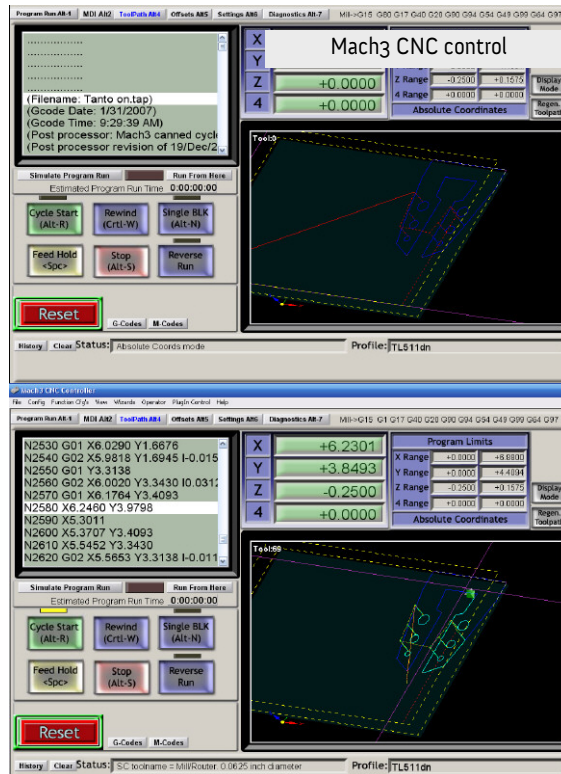
Last thing to do is to run it through a Post Processor* to generate your G-Code!. * A post processor is a configuration file that defines the G-code parameters specific to your CNC Software. In my case, I am using the Mach3 post processor. (Actually called Mach2 due to the first release of Mach software)

Now it's on to the machine!

...or, if you don't have a CNC machine yet, but you couldn't help reading along, there is a nifty free program out there called MicroTech CNC Simulator. (Free!!) <http://www.cncsimulator.com/>



Alternatively, if you don't have a CNC Machine of your own, you can take the G-Code to a CNC Fab shop to have them create your widget for you! Please note, Most Waterjets and some Lasercutters do not use G-code, they use the a native DXF file so you can model only to that point and then send the your DXF to cut out for you!



MTThere is your little titan in the actual CNC controller software, Mach 3, ready to be cut!!!

The cut in process!

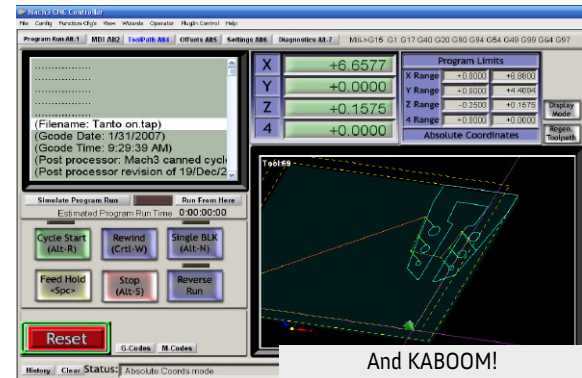
The Yellow 'dashed' outline represents the material.

The Blue is the object to be cut.

The Red is the non-cutting movement of the tool.

The light green is the area that has been cut so far.

The purple gives you reference to the X and Y axis's



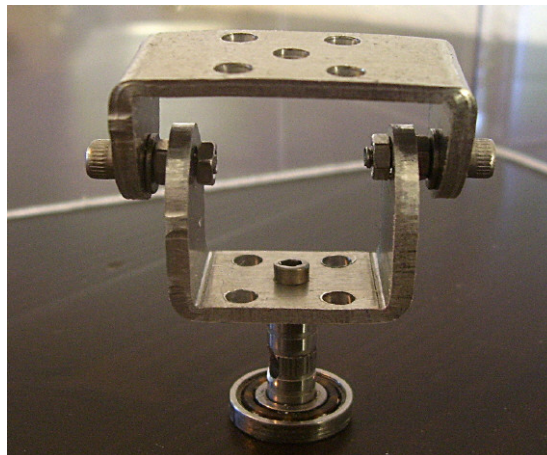
And KABOOM!

You are now a Certified Blender Machinist!

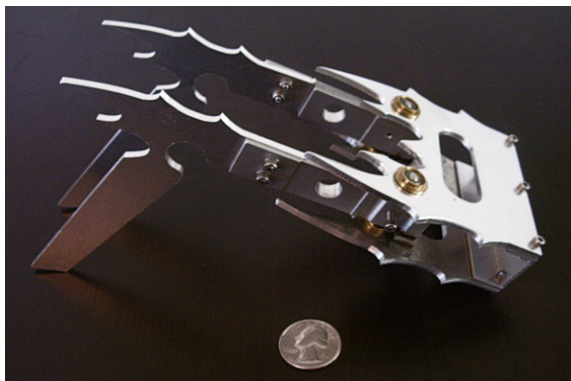
Examples of machined widgets



A single rudder system for a Remote Controlled Boat



A Pan/Tilt mechanism for a tiny Camera (about 1 inch tall)



Dual rudder system for a Remote Controlled Boat



A project in the works! A Remotely controlled 3 wheeled Chopper
This little guy is about 9.5" inches long!

By Michael Kersey

Links to other information

<http://www.wikkidwidgets.com/> - My web where you can see much more examples of widgets created using the methodology in this guide.

http://www.becausewecan.org/Blender_to_CNC – Another methodology of going from Blender to CNC

<http://blender.formworks.co.nz/index.pl?p=2> Lobo's Blender page. This is one of the Python scripts out there deal with Stereolithography CAD files (.STL). STL is a file format native to the stereolithography CAD software created by 3D Systems. This file format is supported by many other software packages; it is widely used for rapid prototyping and computer-aided manufacturing. STL files describe only the surface geometry of a three dimensional object without any representation of color, texture or other common CAD model attributes.

<http://www.instructables.com/id/Easy-to-build-CNC-Mill-Stepper-Motor-and-Driver-ci/> : Easy to build CNC Mill Stepper Motor and Driver circuits

<http://www.homemodelenginemachinist.com/> : Home Model Engine Machinist, Lots of information on making working miniature engines and motors!

<http://www.piclist.com/techref/io/stepper/wires.htm> : A great resource for understanding Stepper motors so you can pull them out of old equipment and use them in your projects.

<http://en.wikipedia.org/wiki/G-code> : Wikipedia's excellent guide to G-Code.

http://woodgears.ca/gear_cutting/template.html : Superb online Gear creation program. You can create gears here and use them directly in the process illustrated in this guide.

<http://www.linuxcnc.org/> : Huge resource of information using Linux instead of windows for DIY CNC

<http://www.machineshopweb.com/> : Don't have your own CNC machine yet? No problem! Huge list of machine shops all over America that can cut out the widgets you design for you! (Note, some shops use a native DXF file in lieu of G-code. If you are going this route, then please check with your local machine shops FIRST!)

<http://www.machinetools.com/mt/> :Need a CNC machine? Huge resource of new and used machine available for purchase! ■

3D WORKSHOP: Create Scenes with a Seamless Background

32



Introduction

In this tutorial, you'll learn how to quickly create scenes with a seamless background suitable for nice studio-styled render of your models, product presentations, cartoon-styled settings and more.

Here are a few examples using this simple setup:



created by Horace Pinker

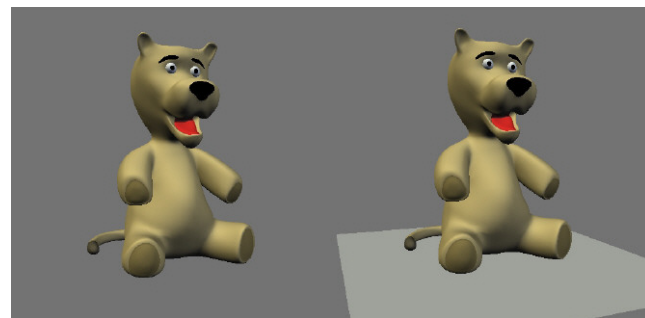


Character created by James Tuvell

How To Setup Your Scene

Setting this up couldn't be easier. Simply add a Plane object that will serve as the floor and set it to only display shadows... that's it!

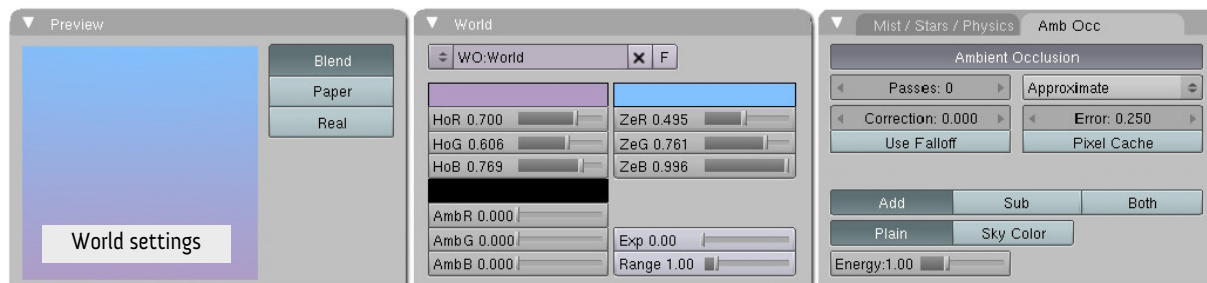
Enable the "OnlyShad" option for the floor object.



By Kernon Dillon

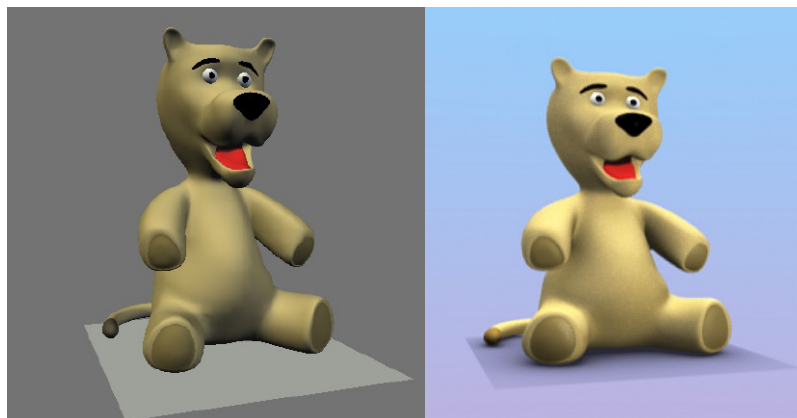
3D WORKSHOP: Create Scenes with a Seamless Background

33



Note that the floor object needs to be large enough to fully capture any shadows cast by the objects in the scene. If the floor object isn't large enough, you'll easily see where the shadows appear to be clipped. Simply increase the floor object's size and the problem is solved.

I'm sure there are many ways to utilize this simple technique to achieve a variety of effects, so have fun and experiment!



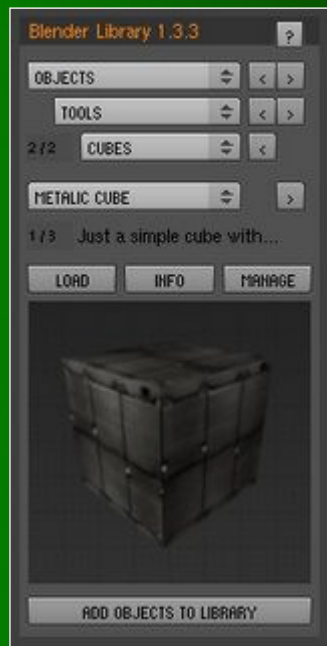
Example of the floor object (Plane) sized too small.

Rendered results of having the floor object sized too small.

Kernon Dillon

Kernon Dillon is a Blender Foundation Certified Trainer and the owner/operator of the BlenderNewbies Video Tutorial website. A free video tutorial demonstrating the technique described in this article can be found at <http://www.blendernewbies.com> [Showcase your 3D Models with a Basic Studio-styled Setup](#)

By Kernon Dillon



by Sandra Gilbert

Personally I hate reinventing the wheel. If I have already created the perfect model, material, lamp setup etc, I really don't want to do it again. I would much rather create new things, not the same thing over and over. Which makes Blender's ability to link/append files one of my all time favorite time saving features. Anything created in one blend file can be used in another blend file. Which gives us the wonderful ability to create reusable Libraries for our present and future projects.

First thing we will want to do is set up and organize our Library so that it is easy and efficient to use. Now everyone has their own way of organizing things, but there are some general organizational ideas that we can take advantage of that will make our Library much easier to use, which is kind of the whole point.

It is highly suggested that you organize your Library into a series of folders. Starting with the obvious main folder "Library" (you can actually name it "Lib" or any other description which works for you, just as long as it is the main folder for the Library). Within that main folder, it is a good idea to set up a series of individual folders to contain all your different types of blender assets that you want to be able to re-use.

Depending on the way you organize things and your particular work flow, you can go with just one level of sub-folders (Materials, Meshes, Textures etc.) or you can create further levels of sub-folders within each sub-folder (ex. In Materials, you could have Wood, Stone, Skin etc.).

Blender's Asset Management

Blender has two options for reusing

blender assets, Linking and Appending (Blender assets can include actions, armatures, cameras, images, IPOs, lamps, materials, meshes, objects, scenes, text, textures, and world, etc.).

Appending will place a independent copy of the material into your new project.

Linking to a material will link to the original file that contains the material, meaning that any changes to the original file will also be saved to the new file that you linked it to. This is a useful option when working on a large project that will generate a large number of files that may be worked on by one or more people. This will allow any changes made during production to be propagated throughout all needed files without the added work of updating each file individually.

Okay, on to actually seeing how to use these options. Both options are started the same way:

- From your new/current project file, either by going to File > Append or Link or using the hotkeys Shift + F1, open the file browser window. All blend files can be used for Appending or Linking.
- Browse to where you have saved the blend file that contains your perfect material.
- Click on the file name of the desired blend file which will open the library list of Appendable/Linkable assets.
- In this case we are looking for materials, so click on Material.
- That will open a list of all available materials in that blend file.
- At the bottom of the file browser there are two buttons, one for Append and one for Link. Push whichever option you have chosen for your project.
- Right Mouse Click on the Material you want to Link or Append, to highlight it and then Middle Mouse click to confirm (load) your material to your new project file.

There, just as I promised, easy as can be. With practice, you won't even have to think about it, you will just click your way to reusing your blender assets.

Granted building a Library won't happen over night and it will always be an ongoing process. As you finish each project, just copy any asset that you think you might use again into the appropriate folder in your Library. To get a jump start on building your library, you can hunt down and collect assets that members of the community have released for use. In fact, if you took the time to gather up all the freely given blender assets floating around here and there, you would have a pretty amazing Library to play with.

For those interested in jump starting or adding to their existing Libraries, here are a few links to some nice collections. Just make sure you pay attention to what copyright licenses are used and any limitations there might be.

[Blender for Architecture](#)

[The Official Blender Model Repository](#)

[Blender Open Material Repository](#)

Once your Library is set up, it is a simple matter of linking or appending the desired asset into your new project. **Appending** is a very straightforward way to get an asset into your new project file as an independent copy of the asset in your Library. Once your appended item is in the new project file, you can continue on your merry way and do what ever you want to it. Life is wonderful!

Linking on the other hand, while it offers some fantastic benefits, can also be confusing unless you are aware of a few things. First the benefits. Linking allows you to have separate files for things like characters, props, environments, etc., that either you or your fellow project members can be working on (or refining) while other things are being done, to say the scene files. Then next time one of you opens the scene file, surprise, your updated character loads from the file that it was linked to. It's a

great way to organize and stream line your work flow. It also makes sure that the latest updates are always loaded and available for whoever needs them.

Now something that you need to be aware of to take advantage of this wonderful thing called **Linking** and it is a biggie, is that Linked objects (assets) **can not** be moved. It is going to stay exactly, and I mean **EXACTLY**, in the same place it occupied in it's original file (the file you linked from). Big bummer to say the least, as well as the cause of many headaches and confusion for artists using the Linking option for the first time.

Right about now you are probably thinking, 'well that isn't useful at all'. Wrong! It is still incredibly useful. You just need to turn your **Linked object** into a **Proxy**. So just what is a proxy?

The most important thing to know about a **Proxy Object** is that it allows you to not only edit local data in your current project file, but more importantly it gives you the option to keep specific data protected. Any data that you set as protected in the original source (Library) file will always be restored from the Library (typically on file reading or undo/redo steps). This protection is set in the linked source (Library) file, which means that only the source (Library) file can determine what can be changed in your current project file.

When working with Poses, (something that you really are not going to want changed in the new project file), you can set the Bone layers as being protected. A protected layer is shown with a black dot in it. Use CTRL+click on a the Bone Layer button to protect or unprotect that layer.

To create a Proxy object in your new project file, Link (*remember Link, not Append*) to your desired object. Once you have your Linked object in your new project

file, select the object and in the 3d view press Ctrl Alt P and confirm the Make Proxy dialog. Your Linked object will be re-named to original name plus a "_proxy" suffix.

Now you can finally edit your Linked object, well within the confines of what has been left unprotected. Depending on what protections were written into the source (Library) file, most likely this will include the ability to change the location and rotation. You also should be able to animate the object's location and animation using lpo Curves. Keep in mind that for mesh objects, the shape of the mesh is what is being protected, so you cannot define new shape keys on the Proxy object. You will have to go back to the source (Library) file for that. Now when you reload your file, Blender will update your Proxy object with any changes made to the original protected data, but will not overwrite any of your local changes (unless the source file was changed to do so).

Note: When selected, Linked objects appear outlined in Cyan, since Proxy objects are now local objects they will be outlined in pink.

We are going to look at one more area of Library use. That is the re-use of Node Layouts. Nodes are a very powerful tool, and like all other blender data can be re-used very easily.

In order to re-use your amazing node layout in a different project, you need to go back to the original (source/Library) file and create a Group for the set of nodes that will be used (don't forget to save the file before you close it). Now when you want to use that node group into your new project, select File->Append and navigate to where you stored it in your Library. When you open the source file's directory file, you will see a NodeTree option. Click it and you will be able to see the list of any available node groups in that file. Select and load the group you want using the usual methods.

For further control of your Library, Mariano [uselessdreamer] has written an amazing Python script called [Blender Library](#), that gives you an easy way to store, manage and retrieve frequently used items; such as materials, textures, objects, etc. There is also built in import and export options to help share your items with the rest of the user community. ■

[Blender Library](#)

[Description of Blender Library Script and Instructions for use.](#)



Introduction

So you have a really cool 3d model that you would love to show off on your web-page. But just how do you go about it? Well, one option would be to take advantage of the fact that 3DMLW now supports the use of blend files.

3DMLW is an Open Source technology for creating and displaying 3D and 2D content on the web through common web-browsers. This technology includes several parts:

- 3DMLW markup language – This is a clear and standardized specification for creating 3DMLW-documents. 3DMLW is similar to XHTML and easy to understand;
- Scripting support for dynamic and interactive content;
- Style sheets for easy and comfortable designing;
- 3DMLW plug-in for web browsers – this is a browser based plug-in for showing 3DMLW documents in web-browsers;
- Editor for 3DMLW – standalone editor & 3DMLW-based web-editor for easily creating 3DMLW-documents.

To see examples of ways to use 3DMLW, be sure to visit their website. Also, they have a nice set of tutorials to get you started on your way to creating some very nice displays of your content.

They also have a new [example](http://www.3dmlw.com/?id=10922) showing skeletal animation in 3DMLW up on their web-site:

<http://www.3dmlw.com/?id=10922>

They have ported 3DMLW to Linux (some minor problems with browser plug-in, but the engine is working). Mac is a bit more complicated, but they already have the rendering window under Mac, too.

Short Tutorial:

With this small tutorial we will show how simple it is to make a 3D model from a .blend file rotate by using 3DMLW.

We'll give a step-by-step tutorial, so you can try to do this yourself by following the steps.

Step 1.

Set the "borders" of the 3DMLW document:

```
<document stylesheet='{#default}'>
</document>
```

Step 2.

Define a stylesheet default and refer to it in the document tag (as is done in Step 1), then define a light in a class light:

```
<stylesheet id='default'>
<class id='light'>
<lights>
<light x='10' y='12' z='25' />
</lights>
</class>
</stylesheet>
```

Step 3.

Define a camera and add a 3D model from the file blendfile.blend in the 3D content that will use the camera and the light defined in the Step 2:

```
<content3d width='100' height='100' camera='{#cam1}'
class='light'>
<camera id='cam1' x='150' z='5' />
<object id='blendfile' x='-15' y='4' z='12' source='blendfile.blend' />
</content3d>
```

Step 4.

Define an animation rotate that rotates the 3D object on the x-axis 360 degrees in 4 second period:

```
<animation id='rotate'>
<key duration='4' angle='360' />
<key duration='0' angle='0' />
</animation>
```

Step 5.

To make the object blendfile rotate by using animation rotate write:

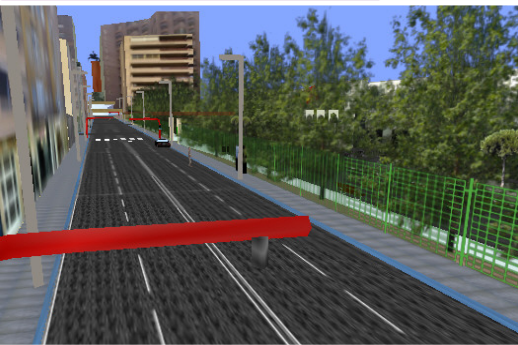
```
<animate animation='{#rotate}' target='{#blendfile}' speed='1'
loop='true' />
```

This will be looped endlessly.

Full source:

```
<document stylesheet='{#default}'>
```

```
<stylesheet id='default'>
<class id='light'>
<lights>
<light x='10' y='12' z='25' />
</lights>
</class>
</stylesheet>
<content3d width='100' height='100' camera='{#cam1}'
class='light'>
<camera id='cam1' x='150' z='5' />
<object id='blendfile' x='-15' y='4' z='12' source='blendfile.blend' />
</content3d>
<animation id='rotate'>
<key duration='4' angle='360' />
<key duration='0' angle='0' />
</animation>
<animate loop='true' animation='{#rotate}' target='{#blendfile}'
interpolation="linear" />
</document>
```



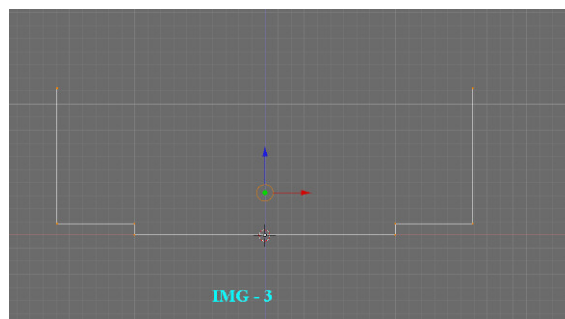
Introduction

Have you ever asked yourself “Why do video games always take place in cities like New York, London, or even Far, far away?”? Would you like to see a video game in your city? Perhaps commercial companies won’t do a game in your city, but you can do it using Blender. All that you need is a computer, Blender, free time and this tutorial. Let’s start!!

a constraint to only move the edges in the Z axis [Mouse Middle Button (MMB)] Now we have something like a U (Img-2). Don’t deselect the new verti-



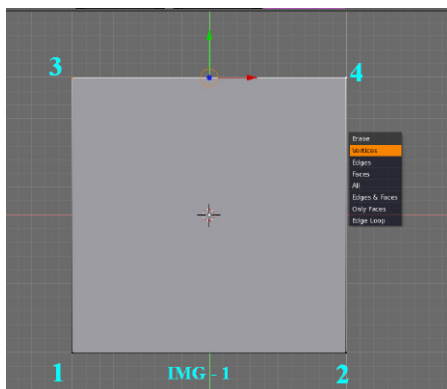
ces and extrude again, but now with a constraint in X axis. Do NOT move the mouse after extruding, just click in order not to move the vertices, then push [S>MMB] and scale that vertices in X axis. Extrude one more time and move the new vertices in the Z axis to have something like (Img-3)



This is the base of our city, when you've done that select all [A>A], go to top view [Pad7] and extrude all [E>Only Edges] don’t forget to add a constraint to only move in Y axis [MMB].

Modelling Basic Streets

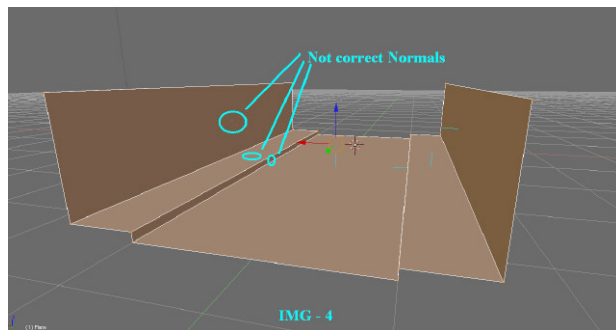
Open Blender and delete the default cube, then add a plane [Add>Mesh>Plane]. Go into Edit Mode [Tab], choose the vertices 3 and 4 and delete them [X key (Erase)>Vertices](Img-1), so we have an edge. Go to side view, [Pad1] choose the other two vertices and extrude them [E>Only Vertices]. While extruding use



by SGMartinez & AARRaiz

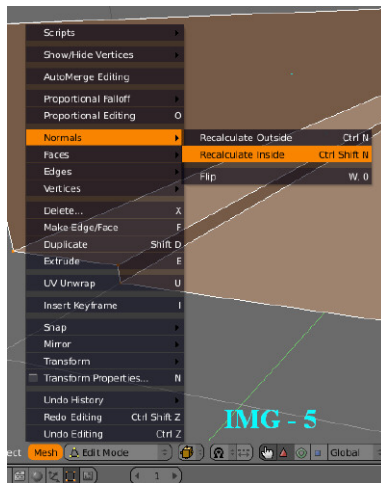
MAKING OF: A Big City For A Game

40

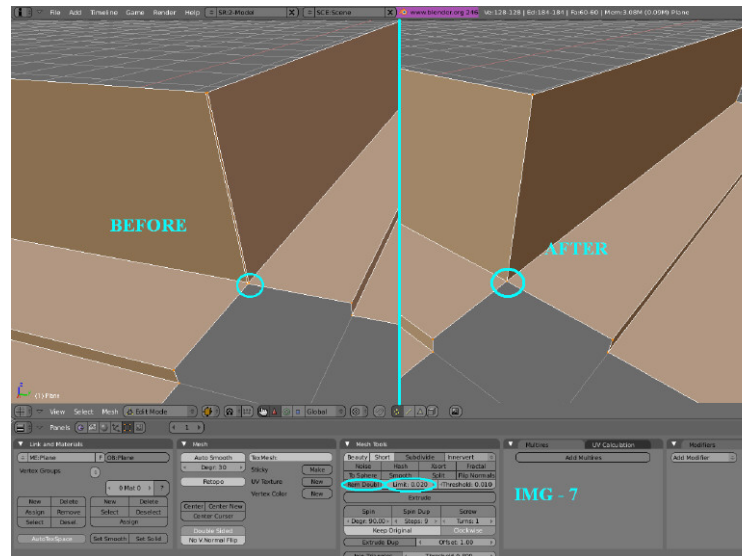
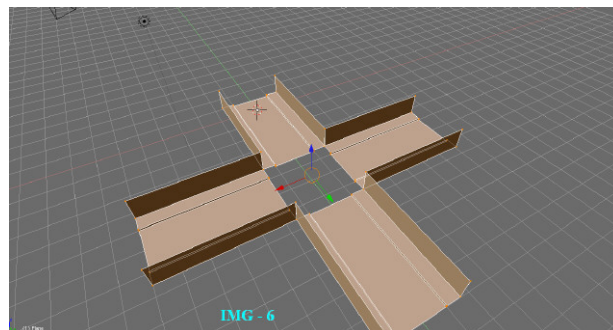


We will have something that will seem like a street, but normals may be flipped, (Img - 4) so recalculate inside (Img - 5) [Ctrl + Shift + N].

Using this method you can do as many streets as you want, use different widths to make little streets or big avenues. Now we're going to model the intersection.

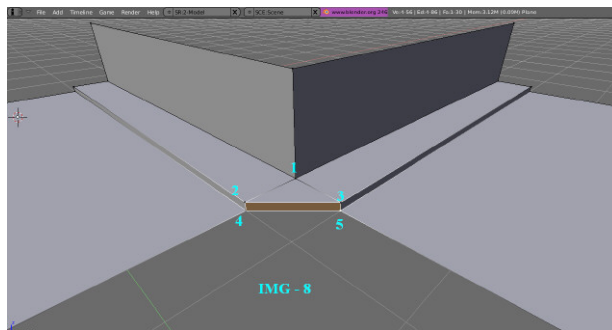


Put 4 streets like in (Img - 6), select all, go to mesh tool and give the limit a value of 0.02, now click on remove doubles. This will join all vertices that are very near (Img - 7).

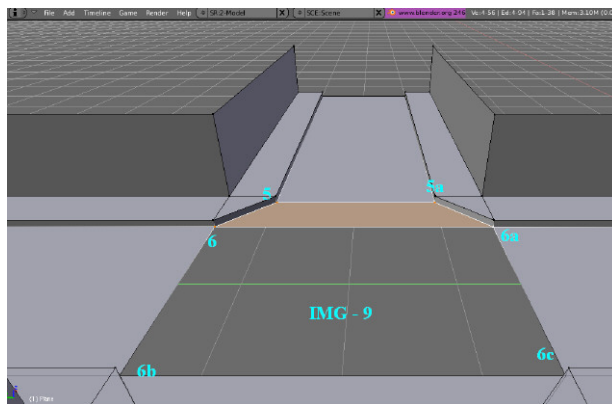


by SGMartinez & AARRuiz

Now select vertices 1, 2 and 3 and push [F]. Select 2, 3, 4 and 5 and push [F] again. (Img - 8). Do the same in the

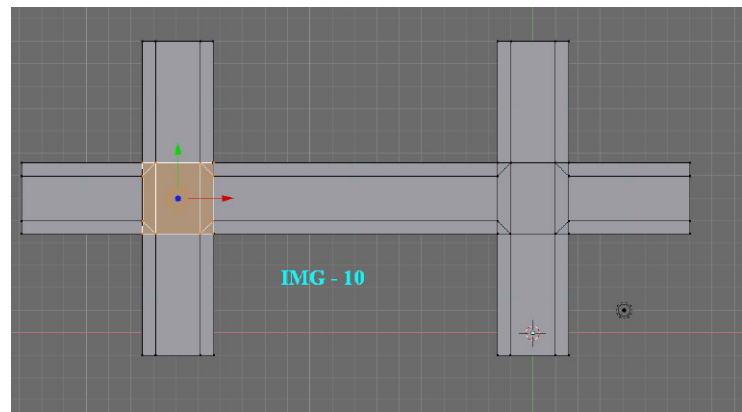


four corners. Then select vertices 5, 6, 5a and 6a and push [F], do the same on the other side and complete selecting vertices 6, 6a, 6b, 6c and pushing [F] (Img - 9). Don't forget to recalculate normals inside if it's needed



Of course, to do more intersections you don't have to repeat all of the process, just select the first corner us-

ing [B] and [Shift + D], put it very near to the new intersection, select all, and click on remove doubles [W>>Remove Doubles] (Img - 10). Doing that, you can make the city as big as you want.



Texturing Streets

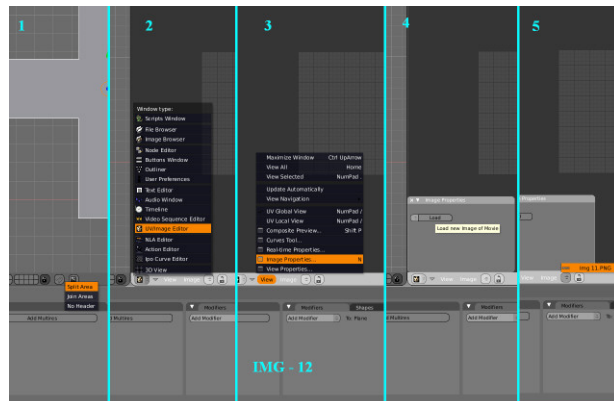
To texture streets you need images of buildings and shops (Img - 11). You can look for them on the internet



or you can go out and take photos, too. Load the image in the UV Image Edit (Img-12)1-push MMB over the line in image and select split areas, 2-change the new window to UV Image Editor, 3-push N, 4-load your image,

MAKING OF: A Big City For A Game

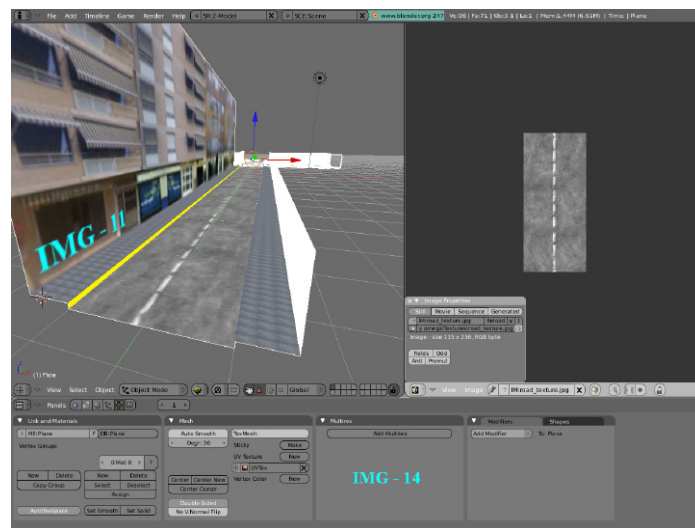
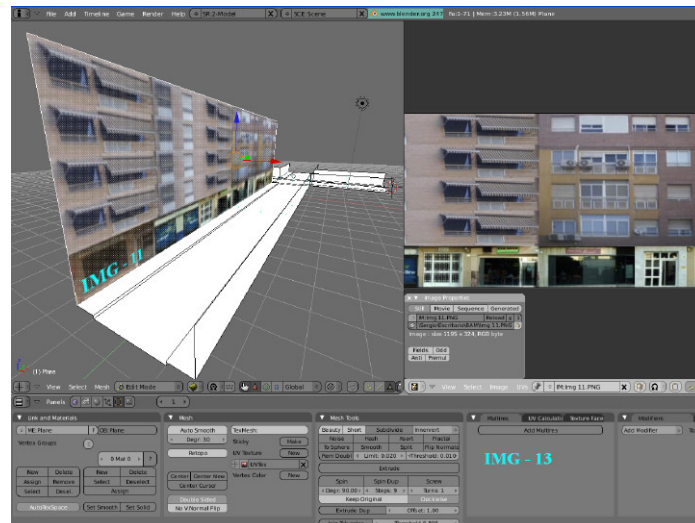
42



5-choose your image in the image browser, go to edit mode [Tab] and change the select mode to faces. Push the face where you want to put the image and choose it in the UV Image Editor (Img-13). Look for a texture of pavement and put it. On the curb put a yellow texture and on the road put other texture something like a road :-D (Img-14)

Adding details

You can use a page like www.3dxttras.com or http://www.katorlegaz.com/3d_models/index.php to look for objects like traffic lights, streetlamp, containers, parking meter. You can use <http://www.google.com> to look for more and more things (Img-15). If you don't find the object you need, model it yourself using tutorials in internet. Add some new cubes to make windows and doors.



by SGMartinez & AARRuiz

Very Big cities

In very big cities you should split the city into some different objects. In each one you have to add a property named “mesh” and the value must be the mesh of the object, an “always sensor” with a delay of ten, a python controller with the script “city.py”.

Finally, add an actuator kind edit object / replace mesh named “m”.

```
#city.py
import GameLogic as GL

list = GL.getCurrentScene().getObjectList()
cont = GL.getCurrentController()
own = cont.getOwner()
d = own.getDistanceTo(list["OBCube"])
m = cont.getActuator("m")
if d > 100 :

    m.setMesh("nada")
else:

    m.setMesh(own.mesh)
GL.addActiveActuator(m, 1)
```

Conclusion

This is the easy way to make your own city and you can add as many objects as you want. When you watch the result you will be proud and you can play all you want now. ■

Happy blending.

Sergio Guevara Martínez (Alias 53R610)

Antonio Alberto Ramos Ruiz (Alias R@MÖ\$)



Introduction

I was inspired by the fantastic matte paintings from the movie Dark Crystal. I've always been a fan of this movie and all the works of Brian Froud and John Howe, so I thought it was both a challenge and real fun to try to mimic those amazing sets. I captured a particular frame I had in my memory, and composed a rough sketch of the image based on it.



Fig 1: A frame from "Dark Crystal"

Modeling

To put it mildly, my modeling skills are very basic. I'm more confident with characters than landscapes or any other thing. So what I did was try to stick precisely to what was going to be seen.

For many elements like the mountains, I just used a mesh patch, roughly modeled, to give the general shape. I then added to it the Subsurf modifier with 4 or 5 subdivisions level and a couple of Displace modifiers using Clouds and Voronoi textures.

The bad thing with Blender is that you can't create a texture without creating a material, so what I do to keep things as tidy as possible is to create the textures I'm going to use in my mesh modifiers, in the same base material the mesh is linked to.

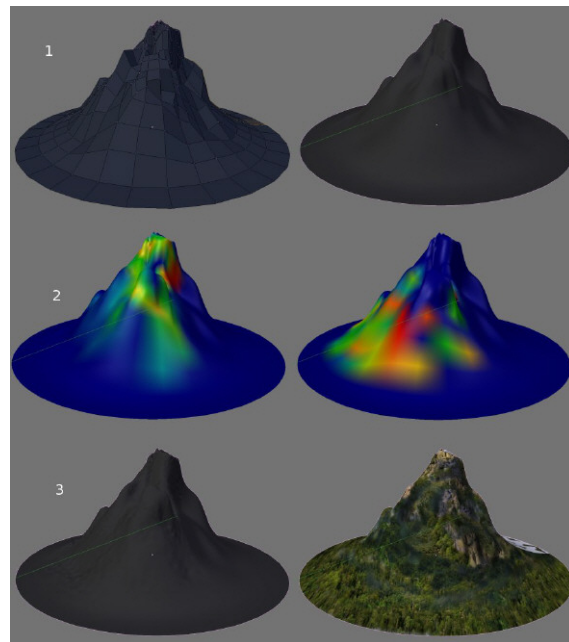


Fig 2: Subsurf is applied on a roughly modelled mountain. 2. Vertex groups controlling effect of displace modifiers. 3. UV texture applied

Then uncheck the textures so they are not rendered.

Last, each Displace modifier acts on the mesh through a vertex group, this gives me more control over which areas were displaced, more or less.

Now the problem with a landscape like this one are the levels of it. I mean, you have got a foreground, which is where the character is and the closer rocks and alien vegetation. Then you have a middle ground, which is the near mountains, the forest and the river. Finally there is a background with a far horizon and the mountains there.

Of course, in the 3D scene all these scales and distances are not existing, and all elements are laid out in a space of around 4-5 blender units. The viewing angle is awkward too. The character is looking down from a very high site, so the horizon line is way up in the image. In this case, you need to do some cheats to mimic the effect of perspective in the horizon, an effect that is not present in a rather small scene like this in 3D.

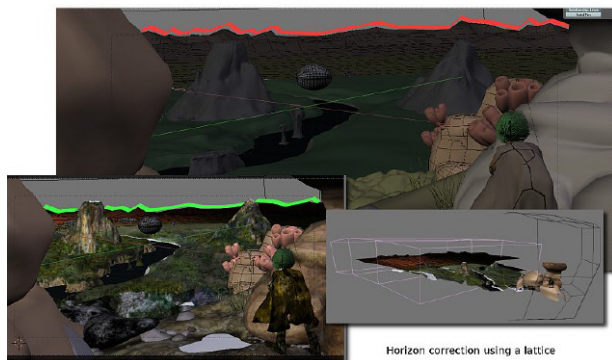


Fig 3: Bending the horizon with a lattice

What I did was using a lattice (yeah, my favorite toy) to curve the distant mountains a little so they stop looking isometric and start looking really far away.

If you ask me for a piece of advice on modeling (mind you!), the only thing I can tell you is to model your objects following the flow or topology needed and try to keep quads as regular as possible. For instance, at first I modeled my mountains out of a square grid. That proved to be wrong when I wanted to get certain natural shapes and volumes, and also when painting textures. It was a lot better (and easier to model) when I started from a circle and modeled the mountain radially. This let me add detail where I needed in a more sensible way. Same applies to rocks and the river.

Also, as a rule of thumb, try to keep your polygons evenly sized, this makes texturing easier and you get very little stretching right from the start.

For the character's poncho, I used a Cloth modifier on a poncho modeled plane. An alpha texture gives the ragged edges. The character was originally modeled in pose, but later on I decided to change the pose for better composition, so I had to rig him in order to change its pose. It was a dirty quick job adding all required bones and a few IK constraints to change his pose.

Texturing

I'd like to say this was the easiest part but I'd be lying. For the forest I tried first using only procedurals combined with displace modifiers to get the look of a heavily dense forest. This worked OK for small renders, but when I went bigger (HD or more) the trick was too obvious, and it looked horrible. So I realized I had to go with more detailed textures. I finally painted a texture using the camera projected unwrap of the forest mesh, using a bunch of photographs from the amazon jungle, rocks,

and a few Gimp filters. I got quite some references from real places like the “Salto del Ángel” in Venezuela, which seemed to fit perfectly for the occasion. I used that for the mountains which are also UV mapped as projected from the camera.



Fig 4: Texture work done over UV layout

Then I added bump maps to un-flatten the forest and then used procedural displacements on top of that on a second Unwrap using regular unwrap methods. This gave me extra details and highlights in certain areas like the shores and cliffs of the river.

Rocks have a similar workflow, I used several procedurals for bump and a UV mapped image for colour. I

Painted the rocks in Gimp on top of the UV layout. For the bump I made a normal map based on the bump map using Gimp's Normalmap filter. It really helps to get some cracks on the rocks.

For the water I just added some procedural bumps and raytracing on a plane. The coral-like alien vegetation uses SSS (subsurface scattering) and a colour texture to make the bands mapped as Tube, nothing really fancy there. I added some bushes using particles just to test the feature.

Lighting

The lighting is done in two stages. First the lamp/AAO thing, then the nodes.

I used a bit of AAO specially for the foreground part, using a World with reddish and blueish tones for Zenith and Horizon. Using the Sky Color option I got some basic tint for the scene.

I then added a spot with irregular shadowmaps for the main shadows in the forest-mountains-river background, plus spots for blue fills to light up the shadows, and yellowish spots for the direct main light.

Additional Lamp lights were added to create fills for the rocks in the foreground that were rendered a bit too dark by the AAO (it renders faster than adding more AAO passes and usually you can have more control over it), and to create back lights for the alien corals which have SSS.

I rendered two main render layers, the “Valley” render layer and the “Rocks” render layer for the foreground. In both cases the usual normal nodes connected to colour-bands do the trick, composing the different normals with Screen/Add or Mix nodes as it best fits. If you are not familiar with this technique it requires you add a

Normal pass to your render layer. Then you connect this Normal pass to a Normal node which lets you control the direction of an intensity gradient. If you also add a colourband you can effectively control the colors of such a gradient.

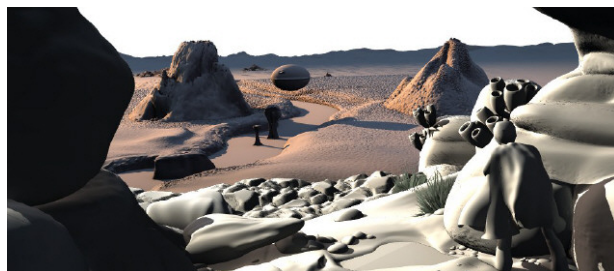


Fig 5: Basic light without Nodes

I used this technique to add highlights and rim lights to the forest and mountains, and also to boost up the blue-ish fills of the mountains and rocks.

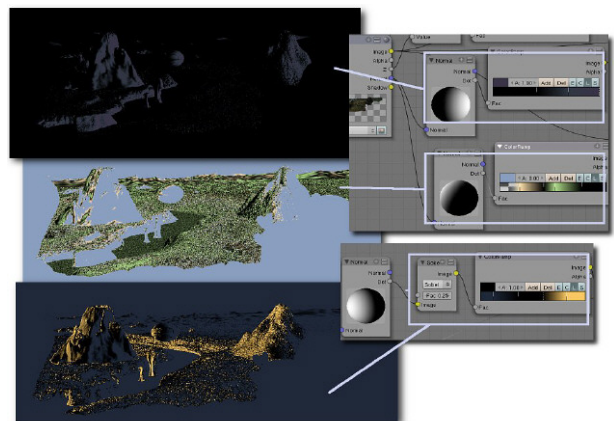


Fig 6: Different fills and backlights with nodes

To mimic Aerial Perspective, I also used a colourband, but this time connected to a Z-Depth pass. This way I could add different colors and intensities for the very far away elements like the horizon mountains and fields. I added different pale blue tones for the mountains and yellowish greens for the near plains.



Fig 7: Faking aerial perspective with Zdepth map and a colourband

For the Sun halo I just added a halo spot on a separate renderlayer and tweaked it as I needed using nodes. This is faster than including the halo in the main renderlayers.

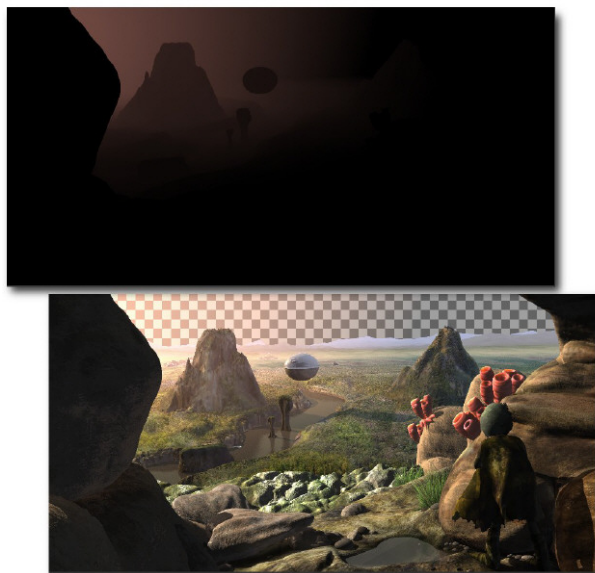


Fig 8: Halo layer and all nodes combined

At this stage I called the lighting done and moved on to compositing.

Compositing

I like to do the compositing in a separate new fresh file. Especially when I have already used Nodes to do the lighting part, it tends to become more messy to my eyes if I keep working on same file for post production.

So, I saved each one of my render layers in a separate file as OpenEXR with Z buffer and also I saved a Multilayer EXR file. This (BIG) file, saves each render layer

without compo nodes applied, and also the separate passes like Normal pass.

So you can have a Z depth map of each layer (included in the single layer EXR files) and Normal pass of each layer as well (included in the Multilayer EXR file).

I also loaded three 2D layers I created in Gimp: Sky gradient, Clouds, and Waterfalls. I just painted them in Gimp on top of a render test with the final resolution as visual reference. It made no sense to do it otherwise and in this way I could control precisely where the clouds should be. Besides, I just love painting clouds, so there they are.



Fig 9: Compositing a waterfall

I wanted to emphasize highlights in the mountains and the forests by adding fine rims, so I started adding a Normal node and processing its output throughout a Sobel filter to create the rims. I also blurred the Normal output and composite again with this Sobel pass. This way you can combine a hard and a soft highlight. Then I added this combined pass to the Valley renderlayer using Dodge filter.

ame technique was applied to the character layer and also to the Clouds 2D layer.

To avoid the edge enhancing to work on the distant mountains in the horizon I used the Z depth to mask it out. You can simply connect the depthmap to a colour-band to get precise control of the intensities and then use this in the Fac socket of the Mix node you are using to blend the layers.

Many times a Map Value node is helpful to rescale the Z depthmap to proper values.

To make things a little dirtier, I added a Blender Cloud texture to the whole image. When all the layers were

composed, I added a RGB node to tweak the overall colour mixing and add some contrast to the image.

Conclusions

You can really go on tweaking colour ad infinitum, and it's really personal taste, so I can't really say I'm totally satisfied with this image, but it certainly helped me to learn new tricks and test different techniques. Hope you guys have found this short summary useful for your own work. ■



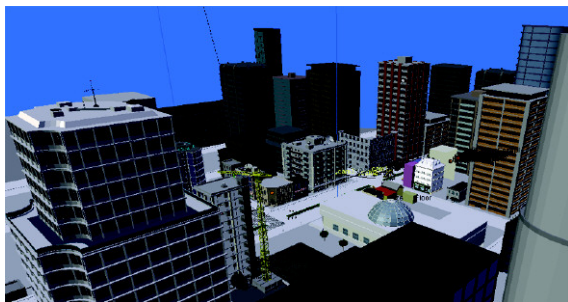
Fig 10: Final render



Introduction

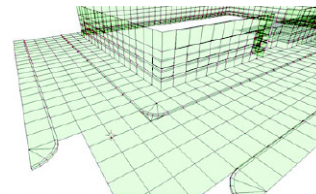
It all started as most projects do, as a simple idea. What I did not realize is that it was not simple. Camera angles, lighting, main characters, background characters, animated props, non-animated props, textures, materials... the list went on and on. What was, and is, in my opinion one of the largest user-based movie projects to ever start, The Crosswalk. I

am going to walk you through the start of the project and show you just how we did what we did so far in creating this huge scene. This article is going to show you some of the many techniques developed for the movie. More precisely, I will be focusing on the building of the scene.



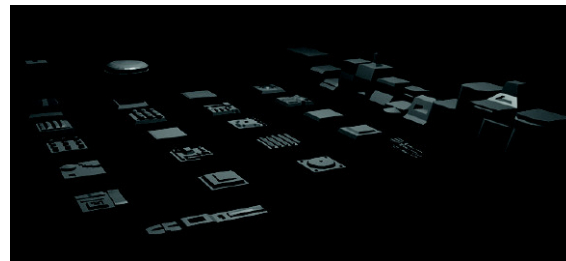
The most updated set for The Crosswalk

When we started working on the base scene (We named it BAS-ESCENE, since it was going to be the base that all the other scenes will be made out of), we started on an extrude and extrude again type of building method for the streets. Once I got to the texturing part I learned very fast that this was not going to work. Here is a very early extrude method example. It looked OK but it was taking me too long to texture it.



Modular Design

I then remembered an old scene I did for a Star Wars project years back. I was building a Death Star scene at the time. I looked at how they built the models for the original movies before they had CG. I was amazed at the simple approach they took. What they did was build a few dozen different molds of panels for the deathstar and cast hundreds of the same ones and just assorted them in a way so as you take a quick look it seems all random and complex.



This example shows how I made all the different panels.

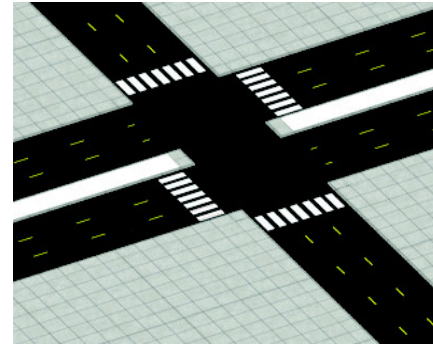
by Enrique Gonzales

MAKING OF: The Scene Of A Dream

51



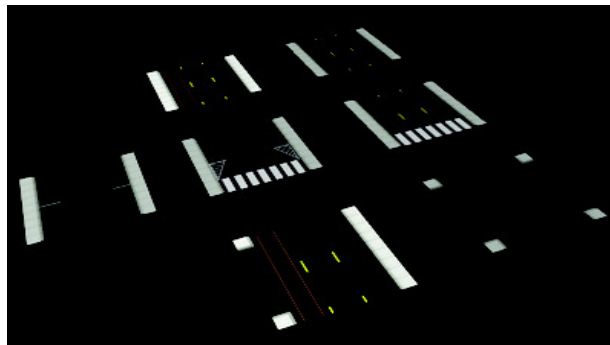
Here you can see them all duplicated and set in random spots to create a very complex look by using a very simple technique.



And here is the final result of duplicating the same tiles over and over.

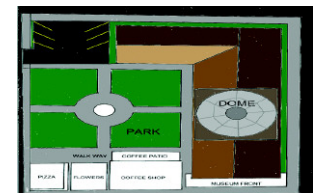
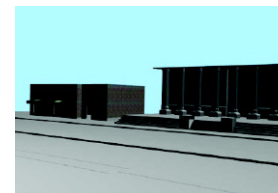
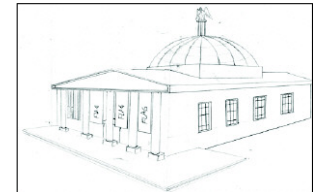
2D to 3D

There was a lot of 2D work done before we started working in 3D. So much that I can only show you a few examples of what we started with. The museum and the coffee shop were the main focus of the movie so they were done first.



I took the same approach to the streets in the Crosswalk

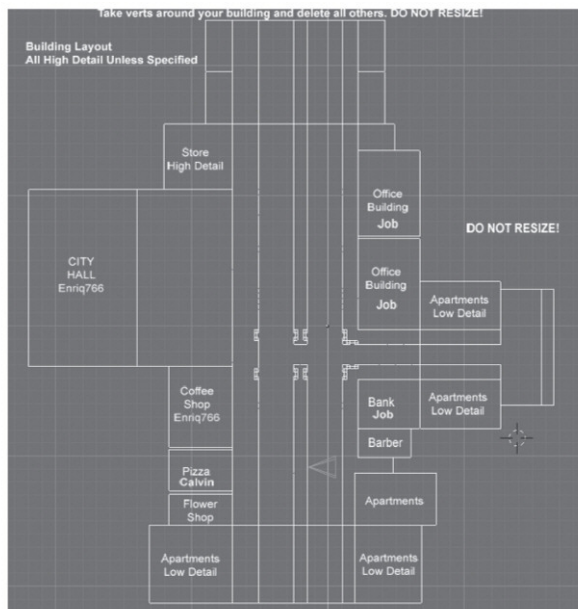
This cut down the texturing time and made creating a whole city very simple.



by Enrique Gonzales

Here is a very early layout map I did to get the idea across all the people that started contributing to the project. It was very crude and I don't think it helped to show them how big each building had to be.

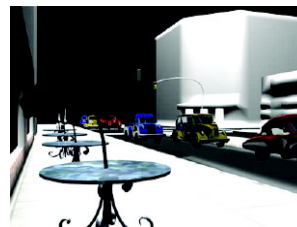
This map was the second version we came up with. It intended to show what was going to be where. After chatting with all the new members of this project we figured we would need more than what was just in front of the camera. The camera will see as far as it can so all those spots and holes needed to be filled. This is where it became complex. From a director's point of view, I tried to visualize what should or should not be in each camera shot. So I came up with this first draft and we started from here.



Cuts And Tests

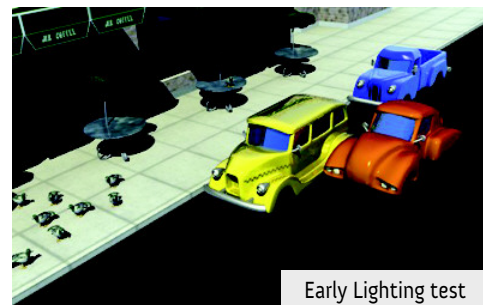
I learned that you have to make tough choices sometimes. This was the first animated test we did using the new cars and tables for the coffee shop. These scenes were to be the final ones for the movie before I decided to kill them and start over. At this point we had the cars a few light poles and the stop light.

The graphic artist (Deadknight) who did the table and stop light has had a huge influence to the overall style of the film. After I saw them and the table, I was so blown away by the look of them that we changed the entire look. First 3D animation test

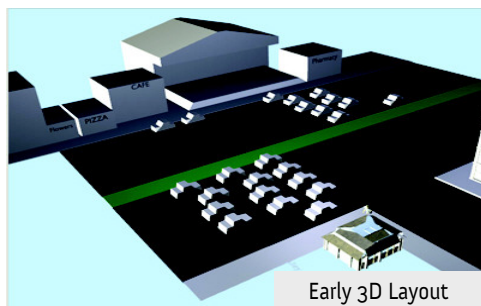


The same goes for the designer of the cars (Sketchy). Once I saw the look of them I knew we had something. Still, it was hard to delete what I had made and spent time on, but in the end it was a decision I do not regret.

This was a very early stage lighting test done with a few of the models we had in hand. In this quick render, the pigeons make their world premiere! These will later be animated by ROUBAL with multiple types of actions like walk, peck, and fly.



At first we had no idea how wide or thin the streets should be. We knew we wanted to have the appearance of a wide and busy street. After we started playing with Henry's walking times we figured out that 3 lanes was as fast as he could walk. This test scene was also scrapped after we went with the modular street design.



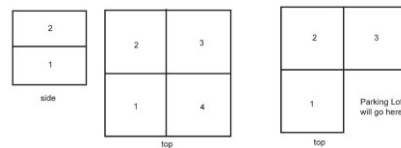
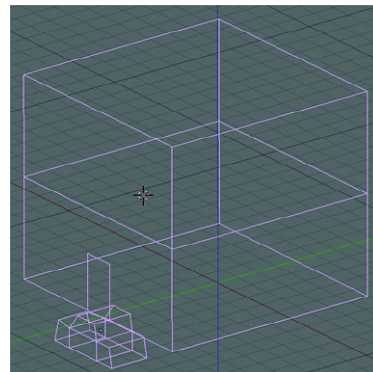
Early 3D Layout

Scale and Rig

Once we started getting dozens of models from the community, I started to append them all into one file. This is where I realized something. I had buildings that just did not match. The scale was off on all of them. Even rescaling did not fix this problem. Also, how do I tell someone who is making a large building to keep in on scale and make it fit in the city block that is designated for it? Well I came up with a very small file called scalerig.blend. It was a cube that could be used to determine the height of a building and the scale of the doors and windows.

The idea was so simple everyone caught on to the idea. I could call for a BANK in the form of a 2x2x2. So the model maker would download the scale rig and duplicate it into 4 pieces and use both 1st and 2nd floors. As long as his building stayed in the bounds of the rig he

was ok. He was free to build as he saw fit with little direction on what could go where. Sometimes I would let them know if it was a corner or street facing building. This would determine where the door would go. With this method we could come up with all kinds of buildings.

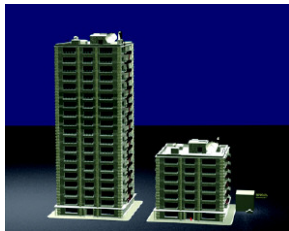


This one was done by Lucid_monkey. This is a very good example of the first models we got after we went to the scale rig system.



I forget who sent this one in - maybe Andy. It was a very detailed building. You could even see the little toys in the window.

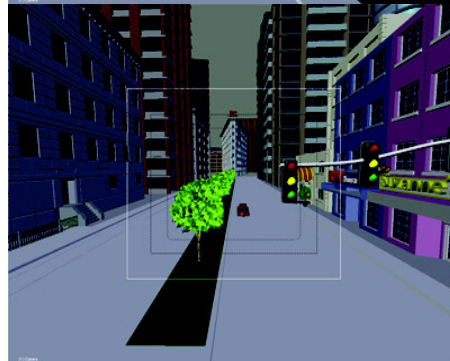
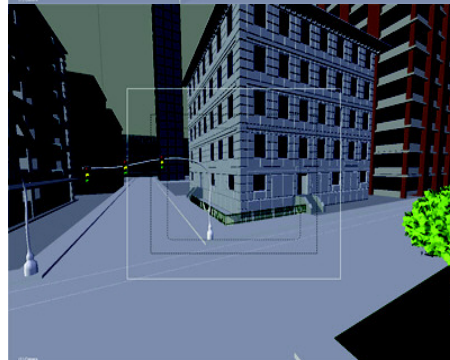
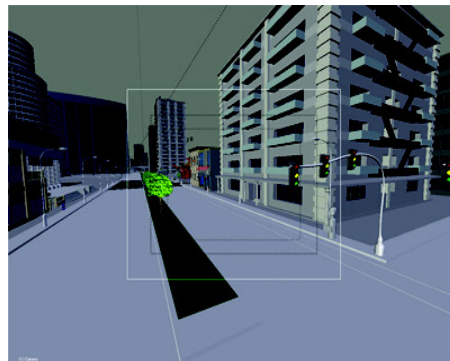
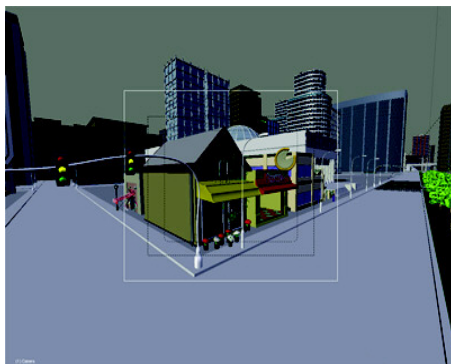




ROUBAL did an awesome job using the scale rig in making some truly amazing sky scrapers. If you look on the bottom, you can see the scale rig.

Putting It All Together

Now that we have all the models and we know where they are going to be placed, it's time to start putting the scene together. All of the closer buildings were made at a higher poly count than the rear ones. These will be in the camera more so we wanted to make them more detailed. I took shots from all angles so you can see we tried to fill all the gaps so it appears the city goes on forever. We also have plans to add in textured buildings in the far off to further give the appearance of a large city.



by Enrique Gonzales

I hope this was an informative article on how we did some of the things for this movie project. I could write an entire book on the things I learned during the making of this film. Most of which are documented in the wiki. Links below. ■

<http://wiki.blender.org/index.php/Tutorials/3DMovies/Index>

<http://www.youtube.com/blenderprojects>

Enrique Gonzales

Enriq766 aka Enriqolonius

Web: www.blenderprojects.org

Email: enriq766@yahoo.com



Sponsors



The BWC 2008: The Results are IN!

This year there were quite a few more entries than last year, and the overall quality was just amazing. Which, of course, made it tougher for the judges to choose just three winners. The results were extremely close and took a while to get narrowed down.

BlenderArt Magazine congratulates not only our BWC 2008 winners, but also the rest of you who showed such great skill and created such a great competition.

The Brief

The brief this year was to create any image, with any theme... with a small twist!

The task was to create a single image that puts us (the viewers) into the middle of a series of events. To tell a story without explanation, and to put the viewers right in the middle.



This year the BWC received massive support and sponsorship with a whopping \$1600-US in prizes and goodies split across four prize pools, one of which will go to the school of the winner's choice.

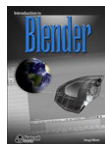
A big thanks to this year's sponsors for supporting the BWC 2008.

Our Sponsors

Natcoll Publishing

Blender E-shop

3Dconnexion



There were an amazing number of quality entries. Be sure to check out all the entries at the home of BWC 2008.

1st Place Soenke (Zordan) - Reaching out

Judges comments

"Incredible detail in the primary and environmental modeling. The character in the foreground is real, she has substance and personality."

"This entry is not only impressive by the huge amount of work it took to complete it, or by its amazing beauty, but also by the epic story it tells in a single image. The way he used a "passive" moment, without any specific action seems to tell a more detailed and rich story than any specific movement or action could have told. Soenke definitely deserves first place for his incredible technical skills. The work he put into modeling all the details of that little city, the care he took to light this immense scene, and his incredible eye for composition brought his rendering to the top of this challenge. Congrats."

"Beautiful work in the style of the traditional romantics. Filled with different symbolic meanings, which makes it very open for interpretation for the viewer. The strong use of perspective and composition, combined with the character in the foreground, makes the work -in my opinion- more about being unreachable (open for interpretation)."

"One of the best Blender images I've seen to date. It is not only technically right, but the story telling is so strong that you can easily be a part of it. You can almost feel the wind blowing. The little birds here and there add a lot to that mood too. The village below with all similar houses but a big one in the middle makes the difference, and there is even life in the dam down there. Very good work on pushing the technical part to the maximum, the lighting is strong too, and the fading-to-blue sky completes the color palette in a gorgeous way. Great achievement Soenke. Strong piece, keep it up!"

"Amazing technique and composition; from an "illustration" point of view, quite a perfect image. I also like the resulting mood."



1st



2nd Place **Andrew (LumpyCow)** - Oh...! That wasn't a good idea

Judges comments

"Bust me up. I literally cracked up (gamering glances from the people in the room). When they saw it, their response told me that this was infectious humor in still life. Keep going with this, it could develop into an awesome short! I'm probably not the only one who wants to see the rest of this story... The originality in developing these two characters is evident. The story jumps out at you. Upon closer inspection, the motion blur and matted fur on the forefeet and paws of the bear shows that it's not all fun and games, and there's a high degree of technical skill displayed as well. Bravo. Please send me the short if it ever comes about!"

"You got awesome expressions there mate! Maybe the environment distracts a little bit because of it's contrast, but maybe you could help it a little by adding a bit of mist or dust from the running fellas? Just an idea. Keep making those crazy animals! =)"

"The image is funny and the toony characters are well made, especially the foreground character (which is also more original than the bear one); and the composition contributes in the right way to the gag."

"I laughed so hard when I saw this one. The expressions on both the bear and the little guy in front are great. It really captures the feeling of the story moment. You can feel the fear from the little guy and you just know that he has serious regrets about messing with the bear. You can all but hear the bear's rage and anger at the little guy. The scene is well modeled and the style fits the story very well."



BWC 2008 :: "Oh! That wasn't a good idea!" :: Lumpy



3rd Place Martin (loramel) - Threesome

Judges comments

"Great story with anticipation, history, consequence. The detail is superb, as are the textures and lighting."

"Out of all the entries, it might be the one that struck me the most as 'A moment in time'. With this snapshot moment from a story of nature, Martin completely nailed the theme. His work on lighting, texturing, and modeling are top notch, but what really stands out here is the composition of the piece. The action can clearly be felt through the picture. Congrats."

"What a 'Moment in Life' for the little mice! It's captured the moment in such a nice way, and the environment also does the job without taking too much attention. Good work on the overall scene."

"I absolutely loved this image. It is beautifully modeled and textured with great attention to detail. You expect to hear the sound of the owl's wings as he is coming in to claim his lunch. You can just feel the tension of the situation jump right out at you as the owl and snake square off over the poor little mouse caught in the sun's rays. I could just feel him hoping for a miracle that the snake and owl would manage to kill each other and save him from being lunch."





Commended Entries

The entries this year were of such a quality that there were a large number of commendations and comments from the judges. A lot of art-work stood out to judges whether through the quality of the storytelling, or the visual impact of the entry. Below are some of the entries that caught the eye of the judges, and the associated comments.

The following are ordered approximately by position and number of comments.

Jack - Meet Me by the Docks





Pablo (pablos) - Hurry, the day is ending!

HURRY UP,
THE DAY IS
ENDING!



Sponsors





Eric (BackiZ) - Meltdown

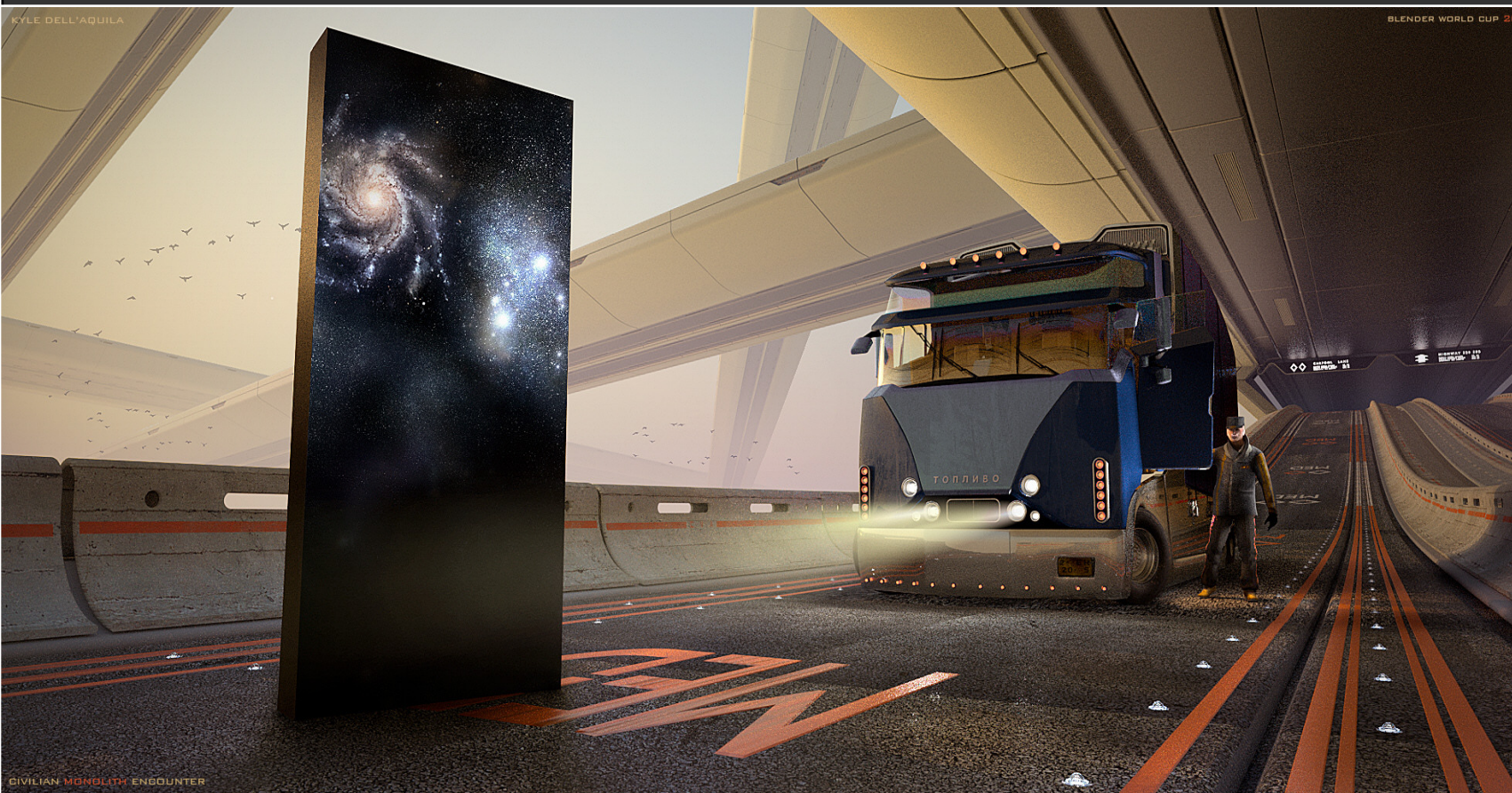


Sponsors





Kyle(shongshong) - Civilian Monolith Encounter



Sponsors





Jakub (Kubeczek77) - Perfect Throw



Blender World Cup 2008

Jakub Batóg

Sponsors





Corey (C. Wynn) - Stonehenge At Twilight



Corey W. Kruger

Sponsors





Mark (mkeller) - The Final Card



Sponsors





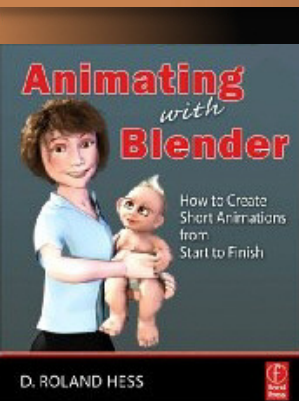
Jared - Now Leaving: Primates



Jared Reissweber 2008

Sponsors





What was the most difficult part in putting this book together?

The hardest part of writing this book was deciding what not to include. In a short animation project, every single part of the process is crucial to a successful outcome, so I wanted to include everything, including the kitchen sink. However, I had a pretty strict target size for the book from the publisher, so I couldn't just write the animation tutorial equivalent of War and Peace either. In the end, I think I came up with a good mix of the essential material, some great tricks and some cool ways of working. The one thing that definitely had to go, though, were detailed step-by-step instructions for many of the more basic operations. This book is not for Blender newcomers. However, if you've mastered the material in *The Essential Blender*, you'll be fine. By the way, if you've watched *The Beast*, you know that the kitchen sink stayed in!

Can one special order an autographed copy?

You can't order one directly from the publisher, but I won't be averse to trying to make it happen through other channels. If you want an actual signed copy, you can contact me directly at animation@harkyman.com. However, one thing that I've seen other authors do is to ask people to send a bookplate along with a self-addressed stamped envelope. A bookplate is an adhesive-backed sheet that usually has some sort of scroll-work or fancy border on it, often with a notice such as "From the library of..." inscribed as well. Then, the author writes something nice on the bookplate, returns it in the provided envelope, and the book owner affixes it to the inside front cover. Not quite the same thing, but it's easy and very inexpensive for everyone involved. If you're interested, contact me at the above email address, and I'll tell you where to send it.

The cool thing about bookplates are that they will often work for any of your favorite authors (living,

obviously)! Just include a nice note to them about how you loved their book, and they will usually be more than happy to inscribe the bookplate and return it to you. It only takes them a second. Make sure you include enough return postage on the enclosed envelope.

What is the greatest change to your Blender work flow (and creative work flow in general) resulting from completing this project?

I will certainly spend more time on my project with rig testing. I mention in the after word of the book that I broke my own rules several times during production, and this was one of them. The rigs for the dogs were pathetically basic, and ended up being more of a hindrance than a help. I was fighting them, and didn't realize their inadequacies until it was way too late. I think that I had just given them a preliminary rig, expecting to come back to it later, but didn't. By the time I understood the damage I had done, I had too many hours into animating them, and too little time left with working on the book and other projects to revisit it. I had to make the best of it.

So, I work my rigs harder now. I was mostly a bare-bones/get-the-job-done rigger, but I'm starting to appreciate the benefits of more complex setups.

How is animating in blender different from animating in other packages?

The only other ways I've animated have been stop-motion and hand-drawn, so I really can't say. It's certainly more efficient than either of those methods!

What are the most important things when learning animation?

I think that the most important thing, and you'll hear this from anyone involved in teaching the arts, is the ability to hone your observational skills. Our memories of motion are imperfect – our brain

creates a shorthand to store what we've seen and experienced. If you only work from that while animating, it's like making a copy of a copy. You need to be able to watch action in real life with fresh eyes, really seeing what's going on. How weight is distributed, how an impact propagates through a body, how someone moves when they are being deceitful, what their body does when they're not thinking about it. Everything. Coupled with that is the ability to observe your own work with a critical eye. Where is the motion wrong? Even if it looks believable, does it convey the story as it should?

What are the most important things to remember when animating in general?

Tell the story. Once you get past the workmanlike aspects -- the technical side -- of animation, you really need to make sure that the animation tells the story. I state in the book, and it shows in *The Beast* which is admittedly not the greatest example of character animation in the world, that I would rather watch a short project that tells a good story but has mediocre animation than the other way around. If your story and characters aren't worth investing in, no one will remember them. If they are interesting or amusing, viewers' brains will fill in the gaps in the animation.

Do you read the BlenderArt magazine?

Every issue. Although I'll admit that sometimes I download it to my desktop and don't get to read it for a day or two. I like to go through magazines all in one sitting, for some strange reason, so I need a nice block of time.

Why do you like to animate?

Hmmm. I like having animated. While I'm doing it, it's a blast. I do not look forward to it, though. I work on so many different kinds of things these days (IT, coding, writing, music and a host of hobbies in addition to animation) that it really takes a serious mental shift to get ready for it, and that's

kind of tough. That said, though, it's strictly etymological. To animate literally means to instill with spirit. Whether it was painting, writing proto-chat-bots in the 80's, creating the AI actors of *Blender-People* or animating, I've always enjoyed creating things that reflect life. And working on the observational techniques that are required to create these things with any kind of effectiveness has paid me back in the other direction as well -- I am more appreciative of the real world after I have tried, pathetically, to recreate it.

Can you describe what could have been a "show stopper" problem while working on this project and how you overcame or solved it?

Well, one of the big points of the book is to build your animation project carefully so that there are no show stoppers. You don't move on to investing more time into a stage until you're sure that the groundwork has been properly laid, and you won't be wasting your efforts. Other than the dog rigs that I mentioned previously, the one disappointment I had was the incompatibility of the cloth simulator with linked library characters. The mother's skirt was going to be a cloth simulation, but the barriers to gracefully integrating it into a large project that used linked characters instead of local assets were too large. I had to re-model, re-rig and ditch the cloth about a third of the way into the actual animation work. Fortunately, I had followed my own rigging (and other) advice with the mother character, so this sort of change went smoothly, and I lost no work.

Having already written two Blender books, do you have any plans to write another book, and what topic would you be covering?

I'm taking a break from writing any Blender-related books this coming year. Being on deadline during Thanksgiving and Christmas for two years in a row hasn't made me the most popular dude in my family.

Right now, I'm taking my time working on an exercise and project book for young fiction writers. My daughter is starting to seriously get into writing, and it's instructive watching her develop in that way. Moving back to Blender, though, Ton and I have kicked around some post Blender 2.5 ideas. Certainly, an updated edition of The Essential Blender will be in order (full color this time), as well as some more advanced materials.

Having completed your first animation project, what if anything would you do differently next time, and will there actually be a next time?

As I mentioned before – rigging. More time and effort spent there. Also, I won't be constraining my next animation project to a simultaneous deadline like I did with the book. That was really rough getting both full sized projects done at once. A looser schedule will allow me to spend the kind of time on the character animation that it deserves.

Anything else you would like to share with our readers, about the book, upcoming projects, Blender or yourself in general.

Let's face it: the CG forums of the world are littered with thousands of threads that were once home to enthusiastic animators who were going to create the Next! Great! Collaborative! Short! Animation! Maybe you were a part of one of those teams. Maybe you yourself started one of those threads. The odds are, though, that those projects died not because of anyone's lack of skill or focus, but because the skill sets that people think are most needed when animating are completely different than the skills required to be the producer of a short animation.

Now, Animating with Blender isn't for everyone. The last thing I want is for someone to buy the book at the wrong point in their development as an artist and as a Blender user. That doesn't help anybody. But if you're to the point where you're a

little bored just making cool stills, you've been experimenting with animation and want to tackle a whole project, this really is the guide for you. There is a time-tested system for producing short animations, and ignoring that wealth of hard-won experience will most likely kill your project regardless of how great an animator you might be. And that's the worst outcome possible once you've spent three hundred hours slaving away in front of your monitor: not finishing. That's what my book really tries to prevent. Whether or not you actually pick up Animating with Blender (which of course you should!), make sure that you read up on the entire process somewhere. Don't become one of the "Hey guys I'm doing an animation!" statistics.

The truism that "Once you've done something, you're ready to begin it." certainly applies here. You learn so much the first time you do a project like this, that your initial result is likely to be... sub-standard, even if you manage to finish it. I really hope that the experience as presented in the book can be a substitute for that first rough journey into short animation: pointing out potential pitfalls, suggesting the best ways around them, and making the mistakes so that you won't have to. Of course, there is more to the book than general process information. It's packed with Blender knowledge: production level use of libraries and linking, the compositor, the sequencer, audio syncing, indoor and outdoor lighting schemes, a cool method for walking (not walk cycles), and a bunch of other stuff. And all of it is focused on using those tools in the service of the short animation production pipeline. For example, the section on the compositor and rendering pipeline doesn't explain the different buttons or panels like a reference or basic tutorial. It really goes into how and why to break your shots up into layers, when and how to extract backgrounds, what to light and what not to light, and how to put it all back together in the compositor for both a higher quality final shot and a faster render.

The author's blog on the book's Amazon.com page has a full breakdown of each chapter with their sub-topics. It's extensive. Also, if you haven't watched The Beast, which is the learning project that is developed throughout the book, you can see it at <http://thebeast.harkyman.com>.

One last thing I want to say is that I love to meet Blenderheads, both beginning and advanced. I'll be speaking at both the Ohio Linux Fest (<http://www.ohiolinux.org>) and the Central Pennsylvania Open Source Conference (<http://www.cposc.org>) in October, and it would be blast to get to meet some of you there. Check the websites, and if you're going to be in the area, drop me an email!

Thanks

Roland Hess

Email: animation@harkyman.com







Stephen Keefer



Blender 2.46 powered !



NATURE JEWEL

COPYRIGHT DAN TABONET 2008

<http://tabonet.free.fr>

✉ t3d @ club-internet.fr







AT THE GRAVEYARD

Gian Leirba, 08.22.2008
www.myspace.com/tomb_of_gabriel



Want to write for BlenderArt Magazine?

78

Here is how!

1. We accept the following:

- Tutorials explaining new Blender features, 3dconcepts, techniques or articles based on current theme of the magazine.
- Reports on useful Blender events throughout the world.
- Cartoons related to blender world.

2. Send submissions to sandra@blenderart.org. Send us a notification on what you want to write and we can follow up from there. (Some guidelines you must follow)

- Images are preferred in PNG but good quality JPG can also do. Images should be separate from the text document.
- Make sure that screenshots are clear and readable and the renders should be at least 800px, but not more than 1600px at maximum.
- Sequential naming of images like, image 001.png... etc.
- Text should be in either ODT, DOC, TXT or HTML.
- Archive them using 7zip or RAR or less preferably zip.

3. Please include the following in your email:

- Name: This can be your full name or blenderartist avatar.
- Photograph: As PNG and maximum width of 256Px. (Only if submitting the article for the first time)
- About yourself: Max 25 words .
- Website: (optional)

Note: All the approved submissions can be placed in the final issue or subsequent issue if deemed fit. All submissions will be cropped/modified if necessary. For more details see the blenderart website.

Issue 19

Rigging and Constraints

- Custom rigs (hands, feet, etc.)
 - How to set up and use
 - Faking muscles/skin movement
- Rigging and Armature tools
- Constraints and Modifiers used for rigging
- Non-character uses for rigs and armatures

Disclaimer

blenderart.org does not take any responsibility both expressed or implied for the material and its nature or accuracy of the information which is published in this PDF magazine. All the materials presented in this PDF magazine have been produced with the expressed permission of their respective authors/owners. blenderart.org and the contributors disclaim all warranties, expressed or implied, including, but not limited to implied warranties of merchantability or fitness for a particular purpose. All images and materials present in this document are printed/re-printed with expressed permission from the authors/owners.

This PDF magazine is archived and available from the blenderart.org website. The blenderart magazine is made available under Creative Commons' Attribution-NoDerivs 2.5' license.

COPYRIGHT© 2008 'BlenderArt Magazine', 'blenderart' and BlenderArt logo are copyright of Gaurav Nawani. 'Izzy' and 'Izzy logo' are copyright Sandra Gilbert. All products and company names featured in the publication are trademark or registered trade marks of their respective owners.